AMS Advanced Change Type Details

AMS Advanced Change Type Reference

Version May 23, 2024

AMS Advanced Change Type Reference: AMS Advanced Change Type Details

Table of Contents

What Are AMS Change Types?

Welcome to the AWS Managed Services (AMS) Change Type Reference. Change Types are the method you use when submitting a request for change (RFC) to indicate what change you want and how it should be implemented.

Change types have a four-part classification scheme: category, subcategory, item, and operation, "CSIO" for short. The category and subcategory are higher-level concepts, and the item and operation specify an entity and the operation that is applied to the entity. For example, the change type that creates an EC2 instance has the classification `Deployment | Advanced stack components | EC2 stack | Create`, and the change type that requests administrative access to that instance has the classification `Management | Access | Stack admin access | Grant`. For more information about change types and requests for change (RFCs), see Change management in the *AMS User Guide*.

This document provides a reference for all of the AMS change types. Any request for change (RFC) that you submit to AMS requires that you specify a change type. If none of the existing change types are appropriate for your request, you can use the `Management | Other | Other | Create` or `Management | Other | Other | Update` classifications.

To learn more about using change types, see the following topics in the *AMS User Guide*:

- Understanding Change Types
- Understanding RFCs

For example walkthroughs of each change type, see the **Additional information** section for the change type, Change Types by Classification.

For a comma-separated value file of change types, open this ZIP file: Change type CSV output file (output-12.2023.zip).

> **Note**
>
> At this time, AMS operates in these AWS Regions: US East (Virginia), US West (N. California), US West (Oregon), US East (Ohio), Canada (Central), South America (São Paulo), EU (Ireland), EU (Frankfurt), EU (London), EU (Paris), Asia Pacific (Mumbai), Asia Pacific (Seoul), Asia Pacific (Singapore), Asia Pacific (Sydney), Asia Pacific (Tokyo)

New Regions are added frequently, however all API calls and CLI operations are run out of `us-east-1`. To learn more, see AWS Regions and availability zones.

Change Types by Classification

Current change type categories are Deployment and Mangement. The Deployment category contains change types that provision AMS resources including by copying or cloning. The Management category contains change types that operate on existing resources.

Change Type Categories

Deployment Category

Change Type Subcategories in the Deployment Category

Advanced Stack Components Subcategory

Change Type Items and Operations in the Advanced Stack Components Subcategory

- ACM | Create Public Certificate
- ACM Certificate With Additional SANs | Create
- AMI | Copy
- AMI | Create
- AMI | Create from Auto Scaling Group
- Application Load Balancer | Create
- Auto Scaling Group | Create
- Cache (ElastiCache Memcached) Stack | Create
- Cache (ElastiCache Redis) Stack | Create
- Database Migration Service (DMS) | Create Replication Instance
- Database Migration Service (DMS) | Create Replication Subnet Group
- Database Migration Service (DMS) | Create Replication Task
- Database Migration Service (DMS) | Create Source Endpoint
- Database Migration Service (DMS) | Create Source Endpoint (MongoDB)
- Database Migration Service (DMS) | Create Source Endpoint (S3)
- Database Migration Service (DMS) | Create Target Endpoint
- Database Migration Service (DMS) | Create Target Endpoint (S3)
- DNS (Private) | Create
- DNS (Public) | Create
- DynamoDB | Create from Backup
- EBS Snapshot | Copy
- EBS Snapshot | Create
- EBS Volume | Create
- EBS Volume | Create from Backup
- EC2 Stack | Create
- EC2 Stack | Create (With Additional Volumes)
- Elastic File System (EFS) | Create
- Elastic File System (EFS) | Create from Backup
- Identity and Access Management (IAM) | Create Access Key

- Identity and Access Management (IAM) | Create Account Alias
- Identity and Access Management (IAM) | Create EC2 Instance Profile
- Identity and Access Management (IAM) | Create Entity or Policy (Read-Write Permissions)
- Identity and Access Management (IAM) | Create Entity or Policy (Review Required)
- Identity and Access Management (IAM) | Create Lambda Execution Role
- Identity and Access Management (IAM) | Create OpenID Connect Provider
- Identity and Access Management (IAM) | Create SAML Identity Provider
- Identity and Access Management (IAM) | Create Service-Linked Role
- Identity and Access Management (IAM) | Create Service-Specific Credentials
- KMS Alias | Create
- KMS Key | Create
- KMS Key | Create (Review Required)
- Listener | Create (For ALB or NLB)
- Load Balancer (ELB) Stack | Create
- Load Balancer (ELB) Stack | Create (With Additional Listeners)
- Network Load Balancer | Create
- OpenSearch | Create Domain
- RDS Database Stack | Create
- RDS Database Stack | Create (For Aurora)
- RDS Database Stack | Create DB Subnet Group
- RDS Database Stack | Create from Backup
- RDS Database Stack | Create from Backup (For Aurora)
- RDS Database Stack | Create from Snapshot
- RDS Snapshot | Copy
- RDS Snapshot | Copy (For Aurora)
- RDS Snapshot | Create
- RDS Snapshot | Create (For Cluster)
- Redshift | Create (Cluster from Snapshot)
- Redshift | Create (Cluster Subnet Group)

- Redshift | Create (Cluster)
- S3 Storage | Create
- S3 Storage | Create Policy (Review Required)
- Security Group | Create
- Security Group | Create (Review Required)
- Storage Gateway | Create from Backup
- Tag | Create
- Tag | Create (Review Required)
- Target Group | Create (For ALB)
- Target Group | Create (For NLB)
- VPC | Add Static Route (Review Required)
- VPC Endpoint (Interface) | Create
- VPN Gateway | Create

ACM | Create Private Certificate

Create a private AWS Certificate Manager (ACM) certificate with email or DNS validation. To create a public ACM certificate, use ct-3ll9hnadql9s1.

Full classification: Deployment | Advanced stack components | ACM | Create private certificate

Change Type Details

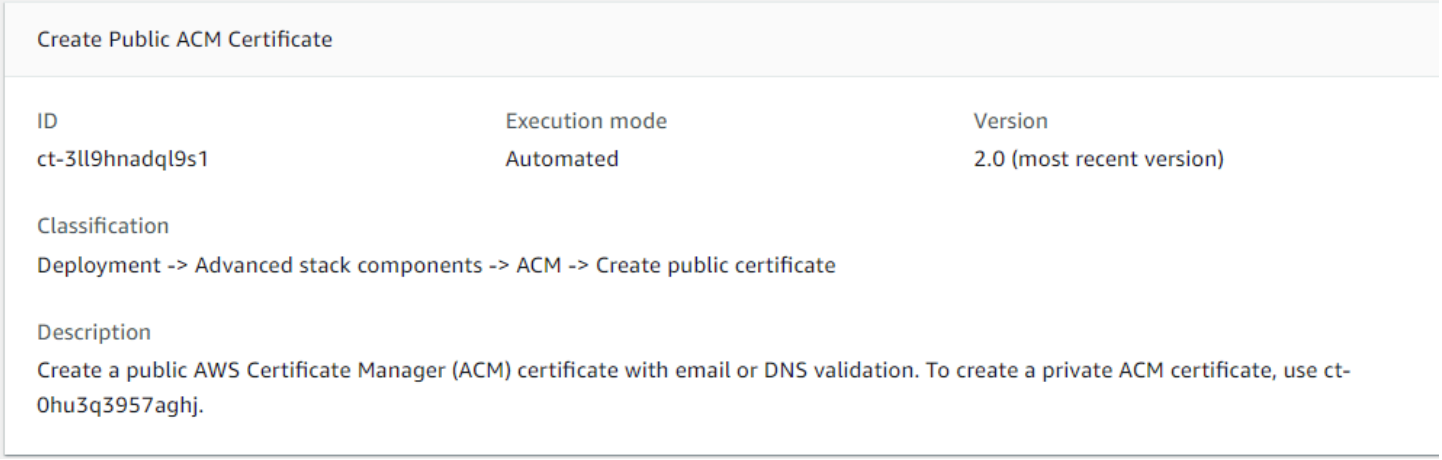
Change type ID	ct-0hu3q3957aghj
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ACM private certificate

Creating a private ACM with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a private ACM with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0hu3q3957aghj" --change-type-version
 "2.0" --title "ACM_PRIVATE_CREATE" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-RequestACMCertificate\",\"Region\":\"eu-west-1\",
\"Parameters\":{\"DomainName\":[\"www.test.com\"],\"CertificateType\":[\"Private\"],
\"Route53DNSValidation\":[\"False\"],\"CertificateAuthorityArn\":[\"arn:aws:acm-pca:eu-
west-1:000000000000:certificate-authority/6a06b611-xxxx-xxxx-xxxx-80cbff8e0000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateAcmPrivateParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0hu3q3957aghj"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateAcmPrivateParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-RequestACMCertificateV2",
       "Region": "eu-west-1",
       "Parameters": {
           "DomainName": [
               "www.test.com"
           ],
           "CertificateType": [
               "Private"
           ],
           "Route53DNSValidation": [
               "False"
           ],
           "CertificateAuthorityArn": [
               "arn:aws:acm-pca:eu-west-1:000000000000:certificate-authority/6a06b611-
   xxxx-xxxx-xxxx-80cbff8e0000"
           ]
   ```

```
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateAcmPrivateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAcmPrivateRfc.json
```

4. Modify and save the CreateAcmPrivateRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-0hu3q3957aghj",
"ChangeTypeVersion":     "2.0",
"Title":                 "ACM-Create-Private-RFC"
}
```

5. Create the RFC, specifying the CreateAcmPrivateRfc file and the CreateAcmPrivateParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAcmPrivateRfc.json  --execution-
parameters file://CreateAcmPrivateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about ACM certificates, see What Is AWS Certificate Manager? and ACM Certificate Characteristic.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0hu3q3957aghj.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-RequestACMCertificateV2",
  "Region": "us-east-1",
```

```
  "Parameters": {
    "DomainName": "www.example-1.com",
    "CertificateAuthorityArn": "arn:aws:acm-pca:us-east-1:000000000000:certificate-
authority/c45863f3-705e-45f6-a3d0-421cf3788800"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RequestACMCertificateV2",
  "Region": "us-east-1",
  "Parameters": {
    "DomainName": "www.example-1.com",
    "CertificateType": "Private",
    "CertificateAuthorityArn": "arn:aws:acm-pca:us-east-1:000000000000:certificate-
authority/c45863f3-705e-45f6-a3d0-421cf3788800",
    "SubjectAlternativeNames": [
      "www.example-1.com",
      "www.example-2.com"
    ],
    "Route53DNSValidation": "False"
  }
}
```

ACM | Create Public Certificate

Create a public AWS Certificate Manager (ACM) certificate with email or DNS validation. To create a private ACM certificate, use ct-0hu3q3957aghj.

Full classification: Deployment | Advanced stack components | ACM | Create public certificate

Change Type Details

Change type ID	ct-3ll9hnadql9s1
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required

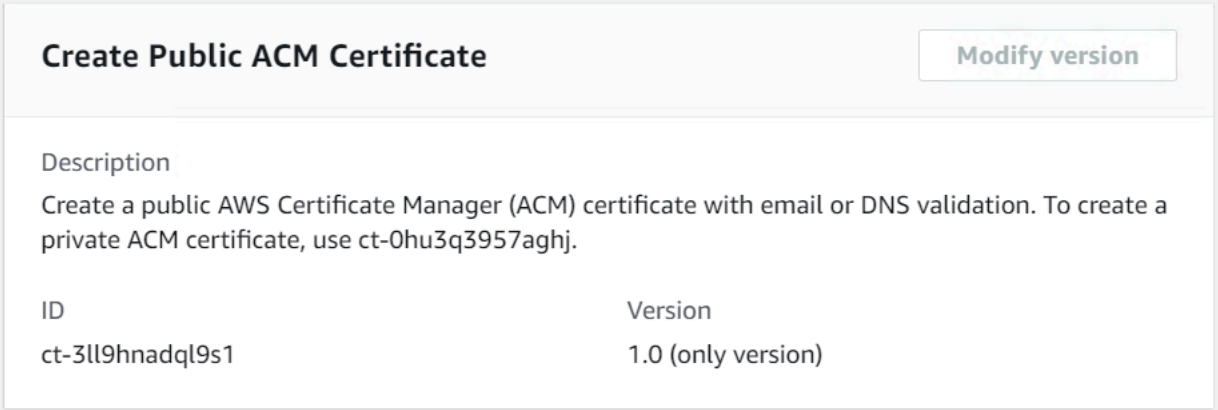
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ACM public certificate

Creating a public ACM with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a public ACM with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3ll9hnadql9s1" --change-type-version
 "1.0" --title "ACM-PUBLIC-CREATE" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-RequestACMCertificate\",\"Region\":\"us-east-1\",\"Parameters
\":{\"DomainName\":[\"www.testing.com\"],\"ValidationMethod\":[\"EMAIL\"],
\"CertificateType\":[\"Public\"],\"ValidationDomain\":[\"\"],\"Route53DNSValidation\":
[\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateAcmPublicParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3ll9hnadql9s1"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateAcmPublicParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-RequestACMCertificate",
       "Region": "us-east-1",
       "Parameters": {
           "DomainName": [
               "www.testing.com"
           ],
           "ValidationMethod": [
               "EMAIL"
           ],
           "CertificateType": [
               "Public"
   ```

```
        ],
        "ValidationDomain": [
            "DOMAIN"
        ],
        "Route53DNSValidation": [
            "False"
        ]
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateAcmPublicRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAcmPublicRfc.json
```

4. Modify and save the CreateAcmPublicRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-3ll9hnadql9s1",
"ChangeTypeVersion":     "1.0",
"Title":                 "ACM-Create-Public-RFC"
}
```

5. Create the RFC, specifying the CreateAcmPublicRfc file and the CreateAcmPublicParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAcmPublicRfc.json  --execution-
parameters file://CreateAcmPublicParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If set to **EMAIL**, ACM sends validation email to the following five common system addresses where *your_domain* is the domain name you entered when you initially requested a certificate and .com is the top-level domain.
>
> - administrator@your_domain.com

- hostmaster@your_domain.com
- postmaster@your_domain.com
- webmaster@your_domain.com
- admin@your_domain.com

If set to **DNS**, ACM provides you one or more CNAME records to add into your DNS database, ACM uses CNAME records to validate that you own or control a domain. If the **Route53DNSValidation** parameter is set to **true** and the ACM certificate and Route53 are in same AWS account, then the CNAME records is added automatically for the validation. If the **Route53DNSValidation** parameter is set to **false** (in the case of a third party DNS Provider), the CNAME records are stored in AWS Secrets Manager. Add the CNAME records to the DNS database manually.

To learn more about ACM certificates, see What Is AWS Certificate Manager? and ACM Certificate Characteristic.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3ll9hnadql9s1.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-RequestACMCertificateV2",
  "Region": "us-east-1",
  "Parameters": {
    "DomainName": "www.example.com",
    "ValidationMethod": "DNS"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RequestACMCertificateV2",
  "Region": "us-east-1",
```

```
  "Parameters": {
    "DomainName": "www.example.com",
    "CertificateType": "Public",
    "ValidationMethod": "DNS",
    "ValidationDomain": "www.example.com",
    "SubjectAlternativeNames": [
      "www.example1.com",
      "www.example2.com"
    ],
    "Route53DNSValidation": "False"
  }
}
```

ACM Certificate With Additional SANs | Create

ACM Certificate with additional SANs

Full classification: Deployment | Advanced stack components | ACM Certificate with additional SANs | Create

Change Type Details

Change type ID	ct-3l14e139i5p50
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ACM certificate with additional SANs

Creating an ACM with the console

The following shows this change type in the AMS console.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an ACM with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

All parameters:

```
aws amscm create-rfc --title test-acm-certificate --change-type-id ct-3l14e139i5p50
 --change-type-version 1.0 --execution-parameters '{ "Description": "Create
 an ACM certificate", "VpcId": "VPC_ID", "Name": "Create an ACM certificate",
 "StackTemplateId": "stm-ftu71ma6q29bvulv0", "Parameters": { "DomainName":
 "*.example.com", "ValidationDomain": "example.com", "SubjectAlternativeName1":
 "*.example-domain.com", "SubjectAlternativeNameValidationDomain1":
 "example-domain.com", "SubjectAlternativeName2": "*.example.net",
 "SubjectAlternativeNameValidationDomain2": "example.net", "SubjectAlternativeName3":
 "*.example-domain.net", "SubjectAlternativeNameValidationDomain3":
 "example-domain.net", "SubjectAlternativeName4": "*.example.org",
 "SubjectAlternativeNameValidationDomain4": "example.org", "SubjectAlternativeName5":
 "*.example-domain.org", "SubjectAlternativeNameValidationDomain5": "example-
domain.org" }, "TimeoutInMinutes": 60 }'
```

Only required parameters:

```
aws amscm create-rfc --title test-acm-certificate --change-type-id ct-3l14e139i5p50
 --change-type-version 1.0 --execution-parameters '{ "Description": "Create
 an ACM certificate", "VpcId": "VPC_ID", "Name": "Create an ACM certificate",
 "StackTemplateId": "stm-ftu71ma6q29bvulv0", "Parameters": { "DomainName":
 "*.example.com" }, "TimeoutInMinutes": 60 }'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateAcmParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3l14e139i5p50" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateAcmParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-ftu71ma6q29bvulv0",
   ```

```
"DomainName":    "DOMAIN_NAME"
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateAcmRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAcmRfc.json
```

4. Modify and save the CreateAcmRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-3l14e139i5p50",
"ChangeTypeVersion":     "1.0",
"Title":                 "ACM-Create-RFC"
}
```

5. Create the RFC, specifying the CreateAcmRfc file and the CreateAcmParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAcmRfc.json  --execution-
parameters file://CreateAcmParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> The timeout setting isn't only about execution, but also your validation of the ACM certificate through email validation. Without your validation, the RFC fails.

To learn more about ACM certificates, see What Is AWS Certificate Manager? and ACM Certificate Characteristics.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3l14e139i5p50.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "DomainName": "example.com",
    "ValidationDomain": "example.com",
    "SubjectAlternativeName1": "domain-1.example.com",
    "SubjectAlternativeNameValidationDomain1": "domain-1.example.com",
    "SubjectAlternativeName2": "domain-2.example.com",
    "SubjectAlternativeNameValidationDomain2": "domain-2.example.com",
    "SubjectAlternativeName3": "domain-3.example.com",
    "SubjectAlternativeNameValidationDomain3": "domain-3.example.com",
    "SubjectAlternativeName4": "domain-4.example.com",
    "SubjectAlternativeNameValidationDomain4": "domain-4.example.com",
    "SubjectAlternativeName5": "domain-5.example.com",
    "SubjectAlternativeNameValidationDomain5": "domain-5.example.com"
  },
  "StackTemplateId": "stm-ftu71ma6q29bvulv0",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

AMI | Copy

Copy an Amazon Machine Image (AMI) in your AMS account.

Full classification: Deployment | Advanced stack components | AMI | Copy

Change Type Details

Change type ID	ct-046aizcwg5idf
Current version	1.0
Expected execution duration	60 minutes

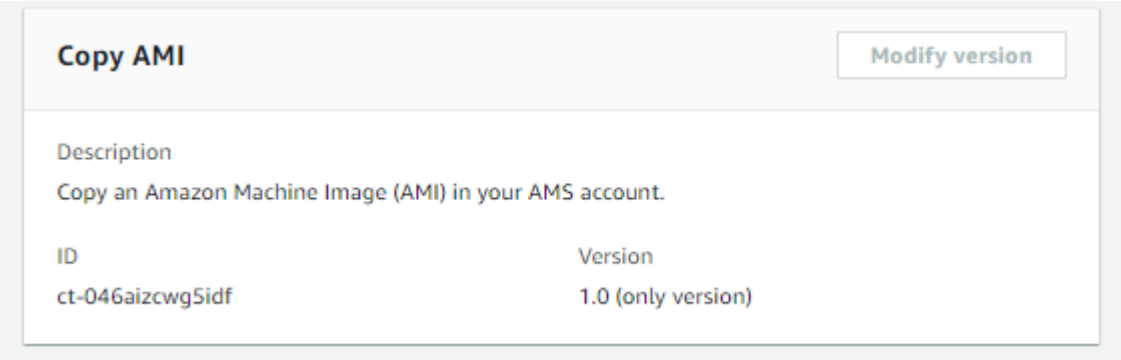
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Copy an AMI

Copying an AMI with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Copying an AMI with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-046aizcwg5idf" --change-type-version "1.0" --
title "Copy AMI" --execution-parameters "{\"DocumentName\":\"AWSManagedServices-CopyAMI
\",\"Region\":\"us-east-1\",\"Parameters\":{\"Name\":[\"AmiName\"],\"SourceImageId
\":[\"ami-1234567890abcdef0\"],\"SourceRegion\":[\"ap-southeast-2\"],\"Encrypted\":
[\"True\"],\"KmsKeyId\":[\"01234567-abcd-abcd-abcd-0123456789ab\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CopyAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-046aizcwg5idf" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CopyAmiParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CopyAMI",
     "Region": "us-east-1",
     "Parameters": {
       "Name": [
         "AmiName"
       ],
       "SourceImageId": [
         "ami-1234567890abcdef0"
       ],
       "SourceRegion": [
         "ap-southeast-2"
       ],
       "Encrypted": [
   ```

```
      "True"
    ],
    "KmsKeyId": [
      "01234567-abcd-abcd-abcd-0123456789ab"
    ]
  }
}
```

3. Output the RFC template JSON file; this example names it CopyAmiRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CopyAmiRfc.json
```

4. Modify and save the CopyAmiRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-046aizcwg5idf",
  "Title": "Copy AMI"
}
```

5. Create the RFC, specifying the CopyAmiRfc file and the CopyAmiParams file:

```
aws amscm create-rfc --cli-input-json file://CopyAmiRfc.json  --execution-
parameters file://CopyAmiParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about copying AMIs, see Copying an AMI.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-046aizcwg5idf.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyAMI",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "Name": [
      "AmiName"
    ],
    "SourceImageId": [
      "ami-1234567890abcdef0"
    ],
    "SourceRegion": [
      "ap-southeast-2"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyAMI",
  "Region": "us-east-1",
  "Parameters": {
    "Name": [
      "AmiName"
    ],
    "SourceImageId": [
      "ami-1234567890abcdef0"
    ],
    "SourceRegion": [
      "ap-southeast-2"
    ],
    "Encrypted": [
      "True"
    ],
    "KmsKeyId": [
      "arn:aws:kms:us-west-2:111122223333:key/01234567-abcd-abcd-abcd-0123456789ab"
    ]
  }
}
```

AMI | Create

Create an Amazon Machine Image (AMI) based on an existing standalone EC2 instance in your AMS account. The instance must be in the stopped state before running this change type.

Full classification: Deployment | Advanced stack components | AMI | Create

Change Type Details

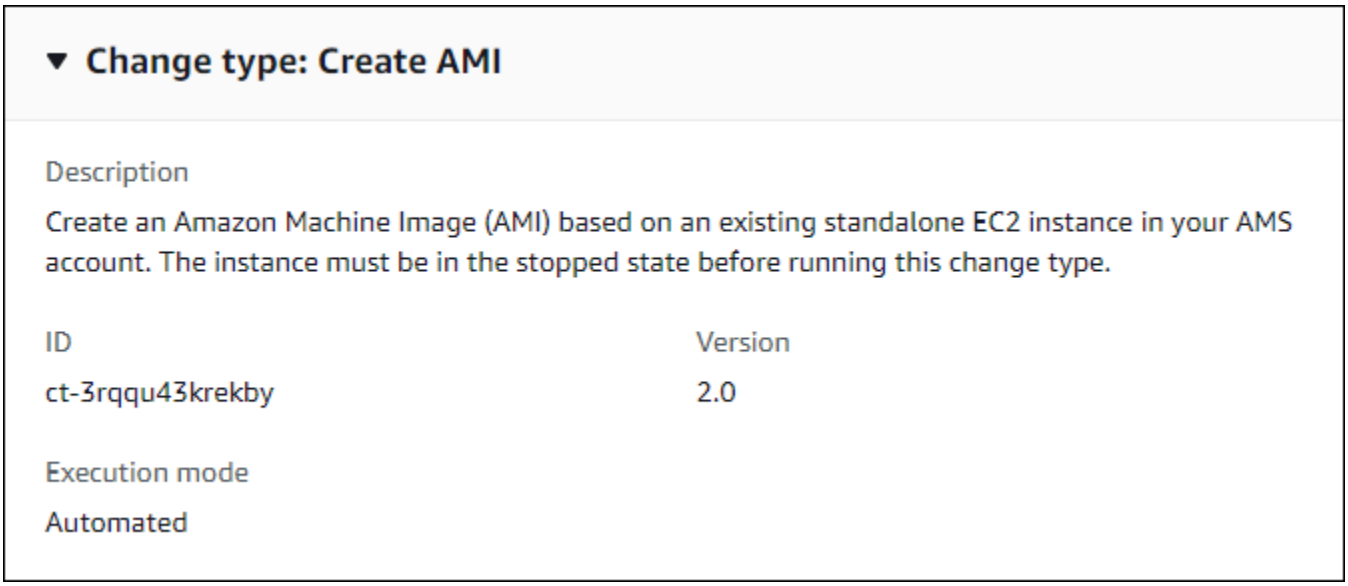
Change type ID	ct-3rqqu43krekby
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create an AMI

Creating an AMI with the console

The following shows this change type in the AMS console.



> **Important**
>
> Before you begin, prepare the EC2 instance that you will use to create the AMI. Without proper preparation, the create AMI RFC is likely to be rejected or fail. For information about preparing your instance to successfully create an AMI, see the instructions included in this tutorial.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AMI with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3rqqu43krekby" --change-type-version "2.0" --title "AMI-Create-IC" --
execution-parameters "{\"AMIName\":\"MyAmi\",\"VpcId\":\"VPC_ID\",\"EC2InstanceId\":
\"INSTANCE_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAmiParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-3rqqu43krekby" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateAmiParams.json
```

2. Modify and save the execution parameters CreateAmiParams.json file. For example, you can replace the contents with something like this:

```
{
"AMIName":          "My-AMI",
"InstanceId":    "EC2_INSTANCE_ID"
}
```

3. Output the RFC template JSON file to a file in your current folder; this example names it CreateAmiRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAmiRfc.json
```

4. Modify and save the CreateAmiRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":         "ct-3rqqu43krekby",
"ChangeTypeVersion":    "2.0",
"Title":                "AMI-Create-RFC"
}
```

5. Create the RFC, specifying the CreateAmiRfc file and the CreateAmiParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAmiRfc.json  --execution-
parameters file://CreateAmiParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> After you have created a custom AMI, you can submit a service request to AMS to have your existing EC2 Auto Scaling group use the new AMI. For information about creating a service request, see Service Request Examples.

> **Important**
>
> Before you begin, prepare the EC2 instance that you will use to create the AMI. Without proper preparation, the create AMI RFC is likely to be rejected or fail.
> To avoid authentication issues from instances created from the new AMI, run these system commands on the instance after applying custom changes, and prior to calling the Create AMI CT.

> **Important**
>
> If the specified instance isn't stopped and separated from its current domain, the AMI creation RFC fails. Prepare the instance as described.
> For more information, see Create a Standard Amazon Machine Image Using Sysprep.

You can subscribe to an AMS SNS AMI notification topic to receive an alert whenever new AMS AMIs of interest to you are deployed. For more information, see AMS AMI Notification Service.

Linux Preparation for AMI Create

Download and run the following script to prepare your instance for AMI creation. You must run this script as the root user.

```
curl https://amazon-ams-us-east-1.s3.amazonaws.com/latest/linux/
prepare_instance_for_ami_and_shutdown.sh -o ./prepare_instance_for_ami_and_shutdown.sh
chmod 744 prepare_instance_for_ami_and_shutdown.sh
./prepare_instance_for_ami_and_shutdown.sh
```

Note: The preceding script performs a shut down on the instance and connected users are logged out from the session.

Windows Preparation for AMI Create

Windows Powershell (run as Administrator):

```
Invoke-AMSSysprep
```

The instance is stopped and any connected user is logged out from the current Windows RDP session.

For more info on creating AWS Windows AMIs, see Create a custom Windows AMI.

UserData for AMI Create

If you need user data to be executed on the next boot from your AMI, ensure the following:

- Registry Key `HKEY_LOCAL_MACHINE\SOFTWARE\Amazon\ManagedServices\RunUserDataViaAMSBootModule` is present; if that key is not present, user data will not be run next time.
- To enable user data to be run on next boot:
  1. Start a Windows PowerShell under administrator privilege (run as administrator)
  2. Run the following command:

     ```
     Install-AMSDependencies
     ```

For information about failed AMI Create RFCs, see RFC failure troubleshooting.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3rqqu43krekby.

Example: Required Parameters

```
{
  "InstanceId": "i-01234567890abcdef",
```

```
  "AmiName": "MyAMI"
}
```

Example: All Parameters

```
{
  "InstanceId": "i-12345678",
  "AmiName": "MyAMI",
  "AmiTags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ]
}
```

AMI | Create from Auto Scaling Group

Create an Amazon Machine Image (AMI) from an EC2 Instance in an Auto Scaling group.

Full classification: Deployment | Advanced stack components | AMI | Create from Auto Scaling group

Change Type Details

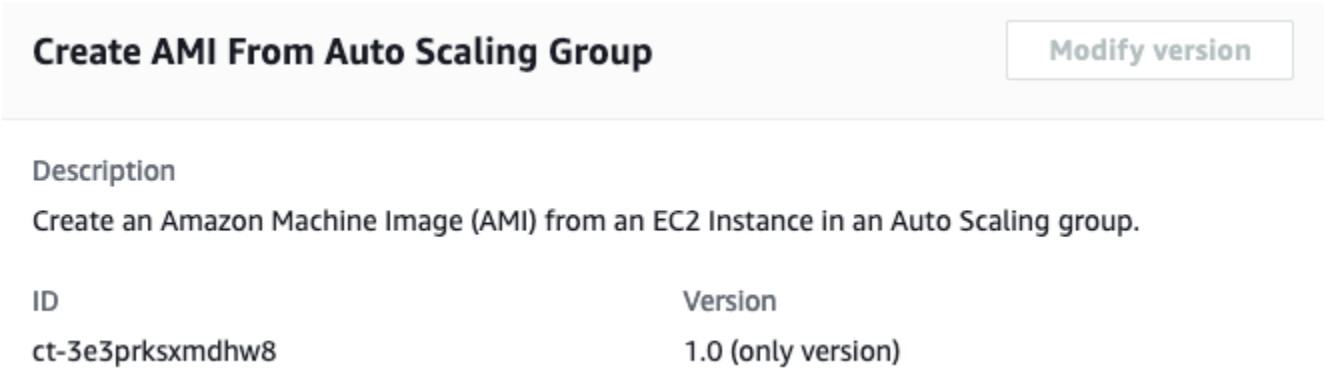
Change type ID	ct-3e3prksxmdhw8
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create AMIs from Auto Scaling groups (ASGs)

Creating an AMI from an ASG with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AMI from an ASG with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3e3prksxmdhw8" --change-type-version "2.0" --title "AMI-Create-IC" --
execution-parameters "{\"AMIName\":\"MyAmi\",\"VpcId\":\"VPC_ID\",\"EC2InstanceId\":
\"INSTANCE_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAmiFromAsgParams.json:

   ```
   aws amscm create-rfc --change-type-id "ct-3e3prksxmdhw8" --change-type-version
    "1.0" --title "Create AMI from an Auto Scaling group" --execution-parameters
    "{\"DocumentName\": \"AWSManagedServices-CreateAmiInAutoScalingGroup\",\"Region
   \": \"us-east-1\",\"Parameters\": {\"AutoScalingGroupName\": [\"stack-ab0123cdef-
   ASG-1ABC2345\"],\"Sysprep\": [\"False\"],\"StopInstance\": [\"False\"]}}"
   ```

2. Modify and save the execution parameters CreateAmiFromAsgParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateAmiInAutoScalingGroup",
     "Region": "us-east-1",
     "Parameters": {
       "AutoScalingGroupName": [
         "stack-ab0123cdef-ASG-1ABC2345"
       ],
       "Sysprep": [
         "False"
       ],
       "StopInstance": [
         "False"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file in your current folder; this example names it CreateAmiFromAsgRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAmiFromAsgRfc.json
```

4. Modify and save the CreateAmiFromAsgRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3e3prksxmdhw8",
  "Title": "Create AMI from an Auto Scaling group"
}
```

5. Create the RFC, specifying the CreateAmiFromAsgRfc file and the CreateAmiFromAsgParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAmiFromAsgRfc.json  --execution-
parameters file://CreateAmiFromAsgParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> After you have created a custom AMI, you can submit a service request to AMS to have your existing EC2 Auto Scaling group use the new AMI. For information about creating a service request, see Service Request Examples.

For information about failed AMI Create RFCs, see RFC failure troubleshooting.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3e3prksxmdhw8.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAmiInAutoScalingGroup",
  "Region" : "us-east-1",
  "Parameters" : {
    "AutoScalingGroupName" : [
      "TestASG"
    ],
    "Sysprep" : [
      "False"
    ],
    "StopInstance" : [
      "False"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAmiInAutoScalingGroup",
  "Region" : "us-east-1",
  "Parameters" : {
    "AutoScalingGroupName" : [
      "TestASG"
    ],
    "Sysprep" : [
      "False"
    ],
    "StopInstance" : [
      "False"
    ]
  }
}
```

Application Load Balancer | Create

Create an AWS Application Load Balancer (ALB), with additional listeners.

Full classification: Deployment | Advanced stack components | Application Load Balancer | Create

Change Type Details

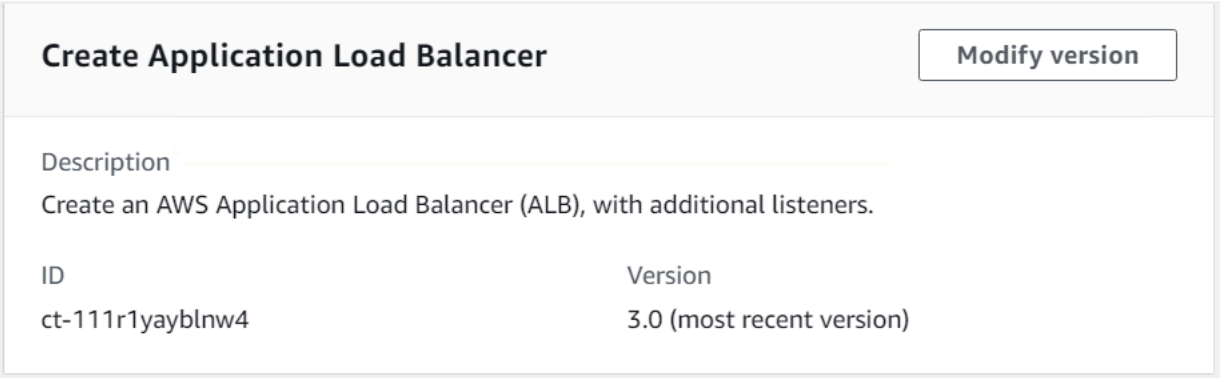
Change type ID	ct-111r1yayblnw4
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create application load balancer (ALB)

Creating an ALB with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an ALB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-111r1yayblnw4" --change-type-version "3.0" --title 'Create ALB' --description
 "My Test ALB"  --execution-parameters ""{\"Description\":\"Test ALB\",\"VpcId\":
\"VPC_ID\",\"Name\":\"TestStack\",\"StackTemplateId\":\"stm-sd7uv500000000000\",
\"TimeoutInMinutes\":360,\"LoadBalancer\":{\"SecurityGroups\":[\"SG_ID\"],\"SubnetIds
\":[\"SUBNET_ID\",\"SUBNET_ID\"]},\"Listener1\":{\"Port\":\"443\",\"Protocol\":
\"HTTPS\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

   ```
   aws amscm get-change-type-version --change-type-id "ct-111r1yayblnw4" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateAlbParams.json
   ```

2. Modify and save the CreateAlbParams file. For example:

   ```
   {
   "Description":      "ALB-Create",
   "VpcId":            "VPC_ID",
   "Name":             "My-ALB",
   "StackTemplateId":  "stm-sd7uv500000000000",
   "TimeoutInMinutes" : 360,
   ```

```
"LoadBalancer" : {
    "SecurityGroups" : ["SG_ID"],
    "SubnetIds" : ["SUBNET_ID", "SUBNET_ID"]
  },
  "Listener1" : {
    "Port" : "443",
    "Protocol" : "HTTPS"
  }
}
```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAlbRfc.json
```

4. Modify and save the CreateAlbRfc.json file. For example:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-111r1yayblnw4",
"Title":                "ALB-Create-RFC"
}
```

5. Create the RFC, specifying the CreateAlbRfc file and the CreateAlbParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAlbRfc.json --execution-
parameters file://CreateAlbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> As of version 3.0, you can also configure four CloudWatch alarms with customized alarm thresholds.

> **Note**
>
> To open ports and associate all the load balancer resources, submit a Management | Advanced stack components | Security groups | Update RFC.

To learn more about AWS Application Load Balancers, see What Is an Application Load Balancer?

To create an Application Load Balancer listener, see Target Group | Create (For ALB).

To create an Application Load Balancer target group, see Create ALB target group.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-111r1yayblnw4.

Example: Required Parameters

```
{
    "Description" : "Test description",
    "VpcId" : "vpc-1234567890abcdef0",
    "Name" : "TestStack",
    "StackTemplateId" : "stm-sd7uv500000000000",
    "TimeoutInMinutes" : 360,
    "LoadBalancer" : {
      "SecurityGroups" : ["sg-1234567890abcdef0"],
      "SubnetIds" : ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
    },
    "Listener1" : {
      "Port" : "443",
      "Protocol" : "HTTPS"
    }
}
```

Example: All Parameters

```
{
    "Description" : "Test description",
    "VpcId" : "vpc-1234567890abcdef0",
    "Name" : "TestStack",
    "Tags" : [
      {
```

```
        "Key" : "foo",
        "Value" : "bar"
      }
    ],
    "StackTemplateId" : "stm-sd7uv500000000000",
    "TimeoutInMinutes" : 360,
    "LoadBalancer" : {
      "Name" : "MyLoadBalancer",
      "SecurityGroups" : ["sg-1234567890abcdef0"],
      "SubnetIds" : ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"],
      "Public" : "true",
      "DeletionProtection" : "false",
      "IdleTimeout" : "60"
    },
    "Listener1" : {
      "Port" : "443",
      "Protocol" : "HTTPS",
      "SSLCertificateArn" : "arn:aws:acm:ap-southeast-2:012345678912:certificate/
e23c3545-e92d-4542-83b8-63483505b5a5",
      "SSLPolicy" : "ELBSecurityPolicy-TLS-1-2-Ext-2018-06"
    },
    "Listener2" : {
      "Port" : "8080",
      "Protocol" : "HTTP"
    },
    "TargetGroup" : {
      "Name" : "MyTargetGroup",
      "HealthCheckInterval" : "10",
      "HealthCheckPath" : "/thing/index.html",
      "HealthCheckPort" : "8080",
      "HealthCheckProtocol" : "HTTP",
      "HealthCheckTimeout" : "10",
      "HealthyThreshold" : "2",
      "UnhealthyThreshold" : "10",
      "ValidHTTPCode" : "200",
      "TargetPort" : "80",
      "TargetProtocol" : "HTTP",
      "DeregistrationDelayTimeout" : "300",
      "SlowStartDuration" : "30",
      "CookieExpirationPeriod" : "20",
      "TargetType" : "instance"
    },
    "HealthyHostsAlarm": {
      "EvaluationPeriods": "10",
```

```
      "Period": "120",
      "Threshold": "1.0"
    },
    "HTTPCodeELB5XXCountAlarm": {
      "EvaluationPeriods": "5",
      "Period": "360",
      "Threshold": "2.0"
    },
    "TargetConnectionErrorsAlarm": {
      "EvaluationPeriods": "5",
      "Period": "360",
      "Threshold": "2.0"
    },
    "RejectedConnectionCountAlarm": {
      "EvaluationPeriods": "10",
      "Period": "120",
      "Threshold": "1.0"
    }
}
```

Auto Scaling Group | Create

Use to create an Auto Scaling group, the launch configuration to use to create new instances when needed, and CloudWatch metrics and alarms for the group.

Full classification: Deployment | Advanced stack components | Auto Scaling group | Create

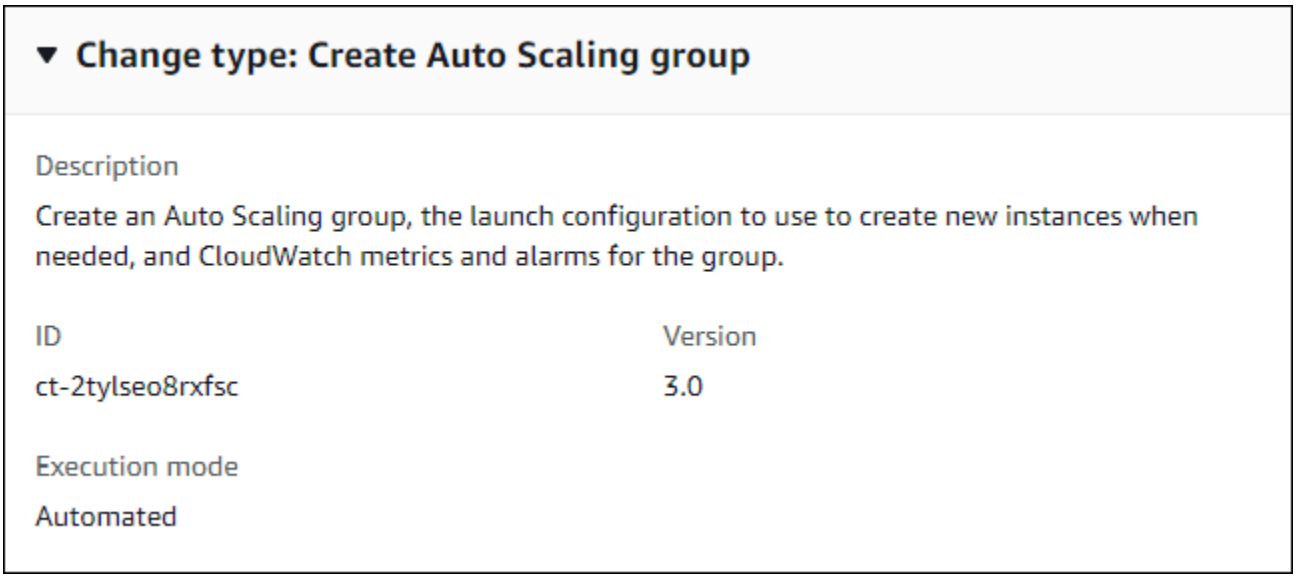
Change Type Details

Change type ID	ct-2tylseo8rxfsc
Current version	4.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create Auto Scaling groups

Creating an Auto Scaling group with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Auto Scaling group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

1. Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example:

   Required parameters only:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-2tylseo8rxfsc" --change-type-version "3.0" --title "ASG-Create-QC" --execution-
parameters "{\"Description\":\"Create a new ASG\",\"VpcId\":\"VPC_ID\",\"Name\":
\"MyASG\",\"TimeoutInMinutes\":360,\"Parameters\":{\"InstanceAmiId\":\"AMI_ID\",
\"InstanceSubnetId\":\"SUBNET_ID\"}}"
```

   Most parameters, except `Tags` and `UserData`:

```
aws --profile saml amscm create-rfc --change-type-id "ct-2tylseo8rxfsc" --
change-type-version "3.0" --title "Stack-Create-ASG-IC" --execution-parameters
 "{\"Description\":\"MyTestASG\",\"VpcId\":\"VPC_ID\",\"StackTemplateId
\":\"stm-suw38u20000000000\",\"Name\":\"MyTestASG\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ASGAmiId\":\"ami-0021317f\",\"ASGCooldown\":300,
\"ASGDesiredCapacity\":1,\"ASGEBSOptimized\":false,\"ASGHealthCheckGracePeriod
\":1800,\"ASGIAMInstanceProfile\":\"customer-mc-ec2-instance-profile\",
\"ASGInstanceDetailedMonitoring\":true,\"ASGInstanceRootVolumeIops\":0,
\"ASGInstanceRootVolumeName\":\"TEST\",\"ASGInstanceRootVolumeType\":
\"standard\",\"ASGInstanceType\":\"m4.large\",\"ASGMaxInstances\":1,
\"ASGMinInstances\":1,\"ASGScaleDownMetricName\":\"CPUUtilization\",
\"ASGScaleDownPolicyCooldown\":300,\"ASGScaleDownPolicyEvaluationPeriods\":4,
\"ASGScaleDownPolicyPeriod\":60,\"ASGScaleDownPolicyScalingAdjustment\":-1,
\"ASGScaleDownPolicyStatistic\":\"Average\",\"ASGScaleDownPolicyThreshold
\":35,\"ASGScaleUpMetricName\":\"CPUUtilization\",\"ASGScaleUpPolicyCooldown
\":60,\"ASGScaleUpPolicyEvaluationPeriods\":2,\"ASGScaleUpPolicyPeriod\":60,
\"ASGScaleUpPolicyScalingAdjustment\":2,\"ASGScaleUpPolicyStatistic\":\"Average\",
\"ASGScaleUpPolicyThreshold\":75,\"ASGSubnetIds\":[\"SUBNET1_ID\",\"SUBNET2_ID\"],
\"ASGUserData\":\"\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CreateAsgParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-2tylseo8rxfsc" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateAsgParams.json
```

2. Modify and save the schema as follows. For example, you can replace the contents with something like this:

> **Note**
>
> Scripts are newline-delimited (separate with literal: "\n"), also, scripts entered as UserData are executed as the "root" user and do not need to use the "sudo" command. The RFC waits up to six hours for all of the UserData script commands to run before returning a final status of success or failure.

```
{
"Description":                          "Create_WordPress_ASG",
"VpcId":                                "VPC_ID",
"StackTemplateId":                      "stm-suw38u00000000000",
"Name":                                 "WP_ASG",
"Tags":                                 [{"Key": "Name","Value": "WP_ASG"}],
"TimeoutInMinutes":                     60,
"ASGAmiId":                             "AMI_ID",
"ASGLoadBalancerNames":                 ["ELB_NAME"],
"ASGSubnetIds":                         ["PRIVATE_AZ1", "PRIVATE_AZ2"],
"ASGUserData":  "#!/bin/bash \n
                REGION=$(curl 169.254.169.254/latest/meta-data/placement/
availability-zone/ | sed 's/[a-z]$//') \n
                yum -y install ruby httpd \n
                chkconfig httpd on \n
                service httpd start \n
                touch /var/www/html/status \n
                cd /tmp \n
                curl -O https://aws-codedeploy-$REGION.s3.amazonaws.com/latest/
install \n
                chmod +x ./install \n
                ./install auto \n
                chkconfig codedeploy-agent on \n
                service codedeploy-agent start"
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; example names it CreateAsgRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAsgRfc.json
```

4. Modify and save the JSON file as follows. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-2tylseo8rxfsc",
"Title":                "ASG-For-WP-Stack-RFC"
}
```

5. Create the RFC, specifying the CreateAsgRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateAsgRfc.json --execution-
parameters file://CreateAsgParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more, see Amazon Auto Scaling.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2tylseo8rxfsc.

Example: Required Parameters

```
{
```

```
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-suw38u40000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ASGAmiId": "ami-1234567890abcdef0",
    "ASGSubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234abcd",
  "StackTemplateId": "stm-suw38u40000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ASGAmiId": "ami-12345678",
    "ASGCooldown": 300,
    "ASGDesiredCapacity": 1,
```

```
    "ASGEBSOptimized": false,
    "ASGIAMInstanceProfile": "customer-mc-ec2-instance-profile",
    "ASGInstanceDetailedMonitoring": false,
    "ASGInstanceRootVolumeIops": 0,
    "ASGInstanceRootVolumeName": "/dev/xvda",
    "ASGInstanceRootVolumeSize": 8,
    "ASGInstanceRootVolumeType": "standard",
    "ASGInstanceType": "m3.medium",
    "ASGLoadBalancerNames": ["elb1"],
    "ASGMaxInstances": 1,
    "ASGMinInstances": 1,
    "ASGHealthCheckGracePeriod": 600,
    "ASGHealthCheckType":"EC2",
    "ASGScaleDownMetricName": "CPUUtilization",
    "ASGScaleDownPolicyCooldown": 300,
    "ASGScaleDownPolicyEvaluationPeriods": 4,
    "ASGScaleDownPolicyPeriod": 60,
    "ASGScaleDownPolicyScalingAdjustment": -1,
    "ASGScaleDownPolicyStatistic": "Average",
    "ASGScaleDownPolicyThreshold": 35,
    "ASGScaleUpMetricName": "CPUUtilization",
    "ASGScaleUpPolicyCooldown": 60,
    "ASGScaleUpPolicyEvaluationPeriods": 2,
    "ASGScaleUpPolicyPeriod": 60,
    "ASGScaleUpPolicyScalingAdjustment": 2,
    "ASGScaleUpPolicyStatistic": "Average",
    "ASGScaleUpPolicyThreshold": 75,
    "ASGSubnetIds": ["subnet-1234abcd", "subnet-1a2b3c4d"],
    "ASGUserData": "pwd\nls -ltrh\necho \"Hello, World\""
  }
}
```

Cache (ElastiCache Memcached) Stack | Create

Use to create an Amazon ElastiCache cluster (one or more cache nodes) that uses the Memcached engine, and specify CloudWatch metrics and alarms for the cluster.

Full classification: Deployment | Advanced stack components | Cache (ElastiCache Memcached) stack | Create

Change Type Details

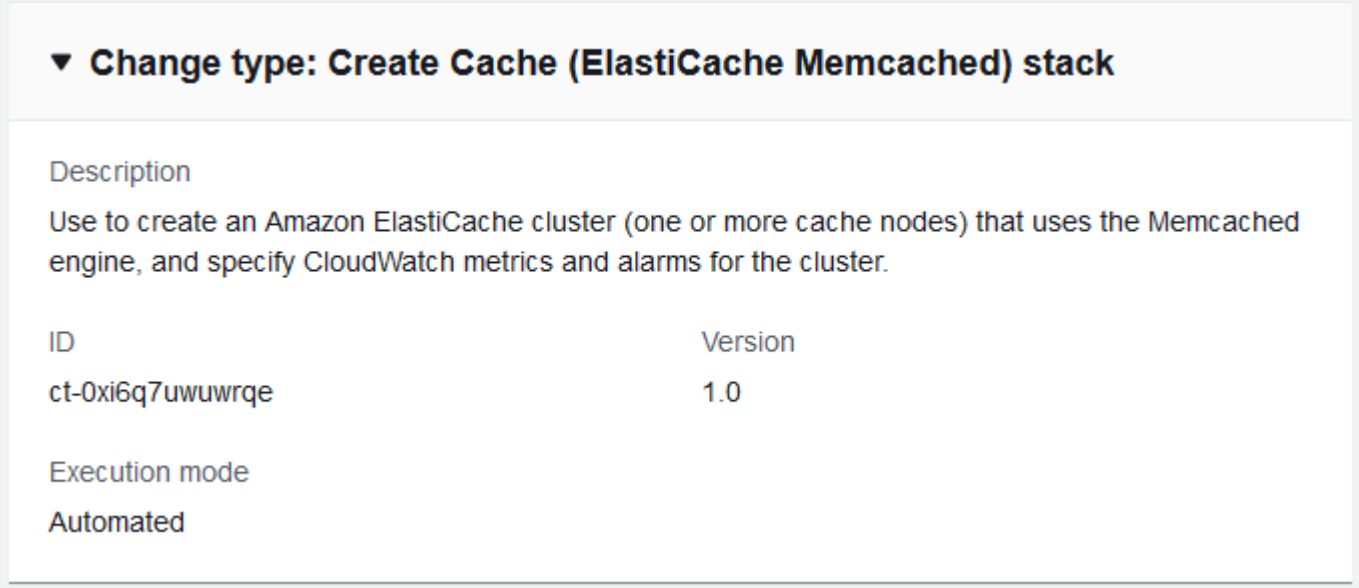
Change type ID	ct-0xi6q7uwuwrqe
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ElastiCache Memcached stack

Creating a Memcached ElastiCache with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Memcached ElastiCache with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0xi6q7uwuwrqe" --change-type-version "1.0" --
execution-parameters "{\"Description\":\"Test description\",\"VpcId\":\"VPC_ID\",\"Name
\":\"TEST_MEMCACHE\",\"StackTemplateId\":\"stm-sfpo2o00000000000\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ElastiCacheAvailabilityZones": [ \"eu-west-1b\", \"eu-
west-1c\" ],\"ElastiCacheClusterName\":\"TEST_NAME\",\"ElastiCacheEngine\":\"redis\",
\"ElastiCacheSubnetIds\":[\"SUBNET_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateMemcacheParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0xi6q7uwuwrqe" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateMemcacheParams.json
   ```

2. Modify and save the CreateMemcacheParams file as follows. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "Description": "This is a test description",
  "VpcId": "VPC_ID",
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheAvailabilityZones": [ "eu-west-1b", "eu-west-1c" ],
    "ElastiCacheClusterName": "test-cluster",
    "ElastiCacheEngine": "memcached",
    "ElastiCacheSubnetIds": [ "SUBNET_ID" , "SUBNET_ID" ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateMemcacheRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateMemcacheRfc.json
```

4. Modify and save the CreateMemcacheRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0xi6q7uwuwrqe",
"Title":                "Memcache-Create-RFC"
}
```

5. Create the RFC, specifying the CreateMemcacheRfc file and the CreateMemcacheParams file:

```
aws amscm create-rfc --cli-input-json file://CreateMemcacheRfc.json --execution-
parameters file://CreateMemcacheParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see Amazon ElastiCache for Memcached.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0xi6q7uwuwrqe.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheAvailabilityZones": [ "eu-west-1b", "eu-west-1c" ],
    "ElastiCacheClusterName": "some-cluster",
    "ElastiCacheEngine": "memcached",
    "ElastiCacheSubnetIds": [ "subnet-1234567890abcdef0", "subnet-1234567890abcdef1" ]
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
```

```
  "VpcId": "vpc-1234abcd",
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheAutoMinorVersionUpgrade": true,
    "ElastiCacheAvailabilityZones": ["eu-west-1a","eu-west-1b"],
    "ElastiCacheClusterName": "mmulti-az",
    "ElastiCacheCPUThresholdAlarmOverride": 95,
    "ElastiCacheEngine": "memcached",
    "ElastiCacheEngineVersion": "1.4.25",
    "ElastiCacheInstanceType": "cache.t1.micro",
    "ElastiCacheMultiAZ": true,
    "ElastiCacheNumberOfNodes": 2,
    "ElastiCachePort": 1121,
    "ElastiCachePreferredMaintenanceWindow": "sun:05:00-sun:09:00",
    "ElastiCacheSubnetGroup": "cachegroup",
    "ElastiCacheSubnetIds": ["subnet-1234abcd","subnet-1a2b3c4d"],
    "SecurityGroups": ["sg-1234abcd"]
  }
}
```

Cache (ElastiCache Redis) Stack | Create

Use to create an Amazon ElastiCache cluster (one or more cache nodes) that uses the Redis engine.

Full classification: Deployment | Advanced stack components | Cache (ElastiCache Redis) stack | Create

Change Type Details

Change type ID	ct-17vnu10suy631

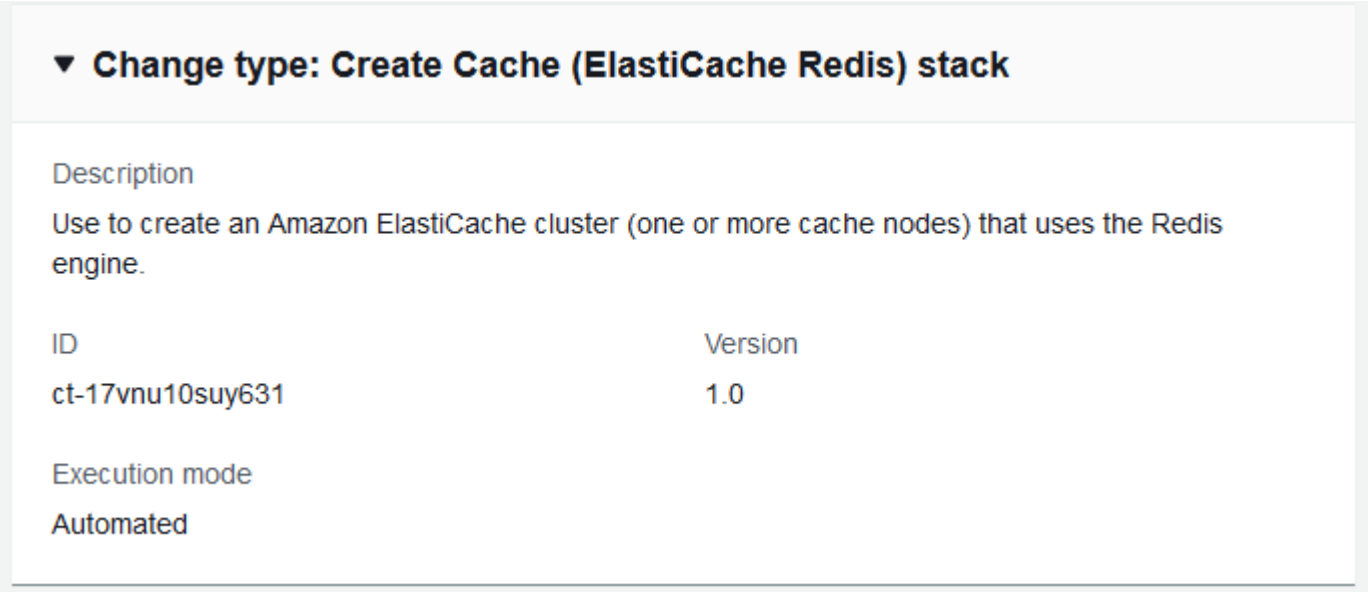
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ElastiCache Redis stack

Creating a Redis ElastiCache with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Redis ElastiCache with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-17vnu10suy631" --change-type-version "1.0" --
execution-parameters "{\"Description\":\"Test description\",\"VpcId\":\"VPC_ID\",\"Name
\":\"TEST_REDIS\",\"StackTemplateId\":\"stm-sfpo2o00000000000\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ElastiCacheClusterName\":\"TEST_NAME\",\"ElastiCacheEngine\":
\"redis\",\"ElastiCacheSubnetIds\":[\"SUBNET_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateRedisParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-17vnu10suy631" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateRedisParams.json
   ```

2. Modify and save the CreateRedisParams file as follows. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "This is a test description",
     "VpcId": "VPC_ID",
   ```

```
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheClusterName": "test-cluster",
    "ElastiCacheEngine": "redis",
    "ElastiCacheSubnetIds": [ "SUBNET_ID" ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateRedisRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRedisRfc.json
```

4. Modify and save the CreateRedisRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-17vnu10suy631",
"Title":                "Redis-Create-RFC"
}
```

5. Create the RFC, specifying the CreateRedisRfc file and the CreateRedisParams file:

```
aws amscm create-rfc --cli-input-json file://CreateRedisRfc.json --execution-
parameters file://CreateRedisParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see Amazon ElastiCache for Redis.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-17vnu10suy631.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheClusterName": "yet-another-cluster",
    "ElastiCacheEngine": "redis",
    "ElastiCacheSubnetIds": [ "subnet-1234567890abcdef0" ]
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-sfpo2o00000000000",
  "Name": "Test Stack",
  "Tags": [
    {
```

```
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ElastiCacheAutoMinorVersionUpgrade": true,
    "ElastiCacheBackupSnapshotRetentionLimit": 5,
    "ElastiCacheClusterName": "project-redis",
    "ElastiCacheCPUThresholdAlarmOverride": 95,
    "ElastiCacheEnableBackup": true,
    "ElastiCacheEngine": "redis",
    "ElastiCacheEngineVersion": "2.8.28",
    "ElastiCacheInstanceType": "cache.t1.micro",
    "ElastiCachePort": 6380,
    "ElastiCachePreferredBackupWindow": "01:00-03:00",
    "ElastiCachePreferredMaintenanceWindow": "sun:05:00-sun:09:00",
    "ElastiCacheSnapshotArns": "arn:aws:s3:::my-bucket/snapshot1.rdb",
    "ElastiCacheSnapshotName": "foo-snapshot",
    "ElastiCacheSubnetGroup": "cachegroup",
    "ElastiCacheSubnetIds": ["subnet-ae7321f7","subnet-05f5cf72"],
    "SecurityGroups": ["sg-4b1b522f"]
  }
}
```

Database Migration Service (DMS) | Create Replication Instance

Create a Database Migration Service (DMS) replication instance on an Amazon EC2 instance in an AMS VPC. Use the replication instance to perform your database migration. The replication instance provides high availability and failover support using a Multi-AZ deployment when you select the Multi-AZ option.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication instance

Change Type Details

Change type ID	ct-27apldkhqr0ol

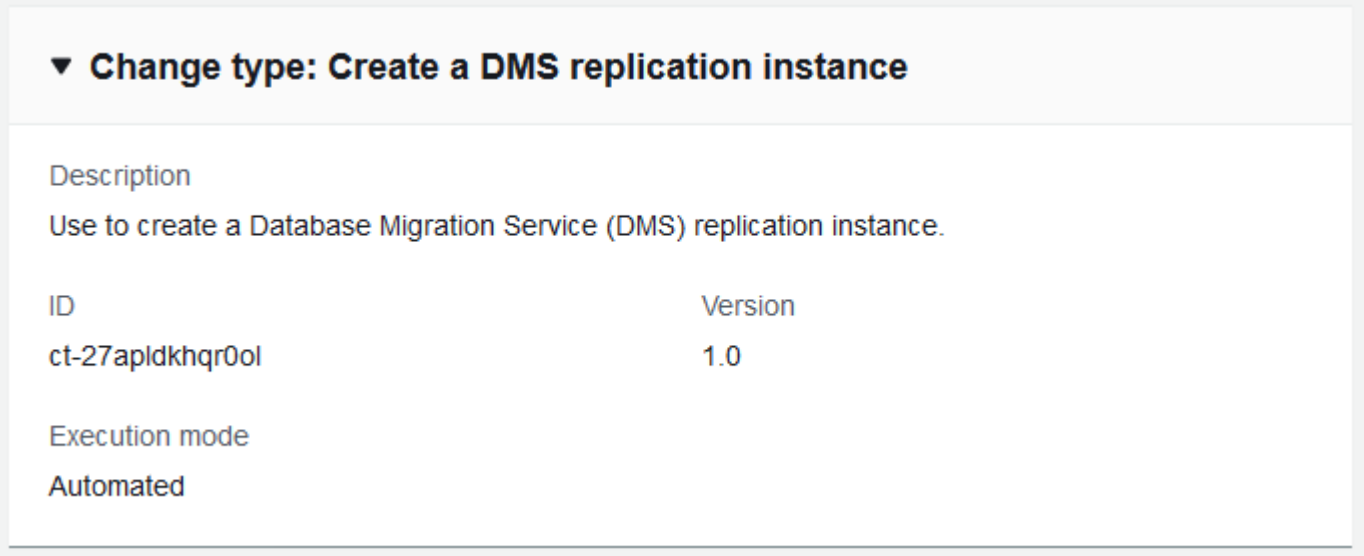
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create AWS DMS replication instance

Creating a AWS DMS replication instance with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a AWS DMS replication instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-27apldkhqr0ol" --change-type-version "1.0" --title "TestDMSRepInstance" --
execution-parameters "{\"Description\":\"DMSTestRepInstance\",\"VpcId\":\"VPC-ID\",
\"Name\":\"REP-INSTANCE-NAME\",\"Parameters\":{\"InstanceClass\":\"dms.t2.micro\",
\"ReplicationSubnetGroupIdentifier\":\"TEST-REP-SG\",\"SecurityGroupIds\":\"SG-ID, SG-
ID\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-3n1j5hdrmiiiuqk6v\"}"
```

While your replication instance is being created, you can specify the source and target data stores. The source and target data stores can be on an Amazon Elastic Compute Cloud (Amazon EC2) instance, an AWS S3 Bucket, an Amazon Relational Database Service (Amazon RDS) DB instance, or an on-premises database.

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateDmsRiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-27apldkhqr0ol" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRiParams.json
   ```

2. Modify and save the execution parameters CreateDmsRiParams.json file. For example, you can replace the contents with something like this:

```
{
"Description":          "DMSTestRepInstance",
"VpcId":                "VPC_ID",
"Name":                 "Test RI",
"StackTemplateId":      "stm-3n1j5hdrmiiiuqk6v",
"TimeoutInMinutes":     60,
"Parameters":   {
    "Description":                      "DESCRIPTION",
    "InstanceClass":                    "dms.t2.micro",
    "ReplicationSubnetGroupIdentifier": "TEST-REP-SG",
    "SecurityGroupIds":                 ["SG-ID, SG-ID"}
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsRiRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRiRfc.json
```

4. Modify and save the CreateDmsRiRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-27apldkhqr0ol",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsRiRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRiRfc.json --execution-
parameters file://CreateDmsRiParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.
- You must create a replication instance on an EC2 instance in your AMS VPC that has sufficient storage and processing power to perform the tasks you assign and migrate data from your source database to the target database. The required size of this instance varies depending on the amount of data you need to migrate and the tasks that you need the instance to perform. The replication instance provides high availability and failover support using a Multi-AZ deployment when you select the `MultiAZ` option. For more information about replication instances, see Working with an AWS DMS Replication Instance.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-27apldkhqr0ol.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "AllocatedStorage": 50,
    "AutoMinorVersionUpgrade": "true",
    "AvailabilityZone": "us-east-1",
    "EngineVersion": "1.5.0",
    "Identifier": "my-instance",
    "InstanceClass": "dms.t2.micro",
    "KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "MultiAZ": "false",
    "PreferredMaintenanceWindow": "sun:06:00-sun:14:00",
    "ReplicationSubnetGroupIdentifier": "my-subnet-group",
    "SecurityGroupIds": ["sg-1234556eaba0a4799", "sg-1234556eaba0a5799"]
  },
  "StackTemplateId": "stm-3n1j5hdrmiiiuqk6v",
  "TimeoutInMinutes": 60,
```

```
  "VpcId": "vpc-01234567890abcdef"
}
```

Database Migration Service (DMS) | Create Replication Subnet Group

Use to create a Database Migration Service (DMS) replication subnet group. Resource creation will fail if the dms-vpc-role IAM role doesn't already exist.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication subnet group

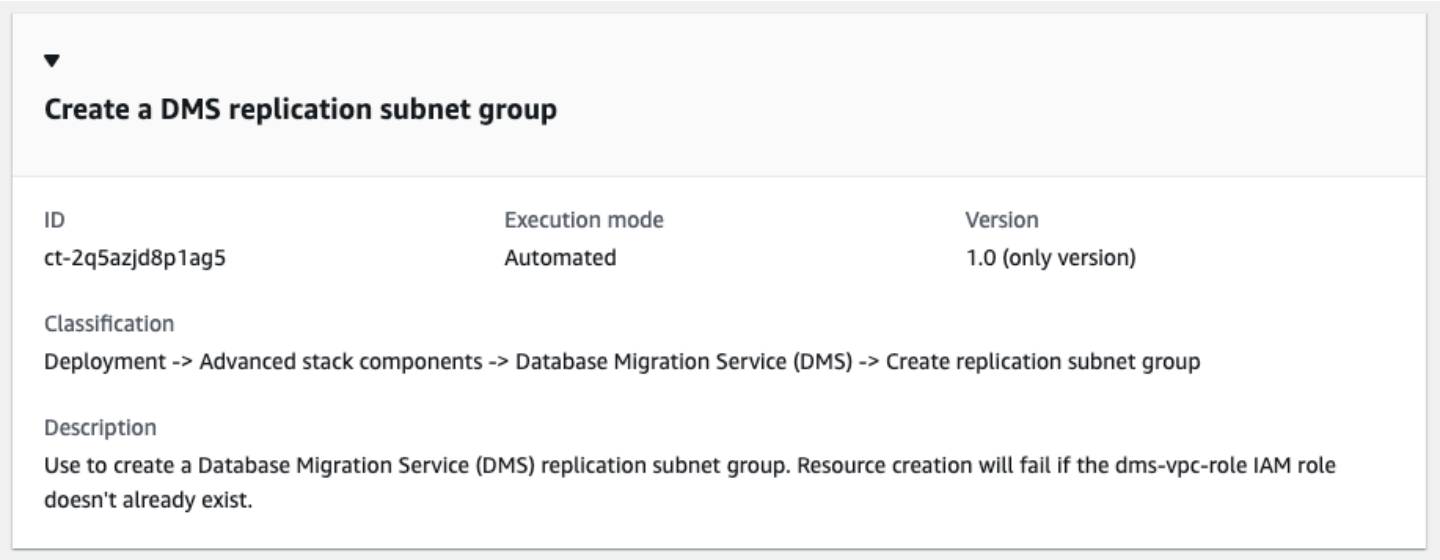
Change Type Details

Change type ID	ct-2q5azjd8p1ag5
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create AWS DMS replication subnet group

Creating a AWS DMS replication subnet group with the console



Note

This CT fails if the `dms-vpc-role` IAM role doesn't exist in the account.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a AWS DMS replication subnet group with the CLI

> **Note**
>
> This CT fails if the `dms-vpc-role` IAM role doesn't exist in the account.

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-2q5azjd8p1ag5" --change-type-version "1.0" --title "TestDMSRepSG" --execution-
parameters "{\"Description\":\"DMSTestRepSG\",\"VpcId\":\"VPC-ID\",\"Name\":\"Test
 Stack\",\"Parameters\":{\"Description\":\"DESCRIPTION\",\"SubnetIds\":[\"SUBNET-ID\",
 \"SUBNET-ID\"]},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-j637f96ls1h4oy5fj
\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateDmsRsgParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2q5azjd8p1ag5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRsgParams.json
   ```

2. Modify and save the execution parameters CreateDmsRsgParams.json file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "DMSTestRepSG",
   ```

```
"VpcId":                "VPC_ID",
"TimeoutInMinutes":     60,
"StackTemplateId":      "stm-j637f96ls1h4oy5fj",
"Name":                 "Test RSG",
"Parameters":   {
    "Description":            "DESCRIPTION",
    "SubnetIds":              ["SUBNET_ID", "SUBNET_ID"]
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsRsgRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRsgRfc.json
```

4. Modify and save the CreateDmsRsgRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2q5azjd8p1ag5",
"Title":                "DMS-RSG-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsRsgRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRsgRfc.json --execution-
parameters file://CreateDmsRsgParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- This CT fails if the `dms-vpc-role` IAM role doesn't exist in the account.
- You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

For more information about DMS replication instances and subnet groups, see Setting Up a Network for a Replication Instance.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2q5azjd8p1ag5.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "Test Stack",
  "Parameters": {
    "Description": "test description",
    "SubnetIds": ["subnet-1234567890abcdef0"]
  },
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-j637f96ls1h4oy5fj"
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "Parameters": {
    "Identifier": "myidentifier",
    "Description": "test description",
    "SubnetIds": ["subnet-12345678"]
  },
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-j637f96ls1h4oy5fj"
}
```

Database Migration Service (DMS) | Create Replication Task

Use to create a Database Migration Service (DMS) replication task.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication task

Change Type Details

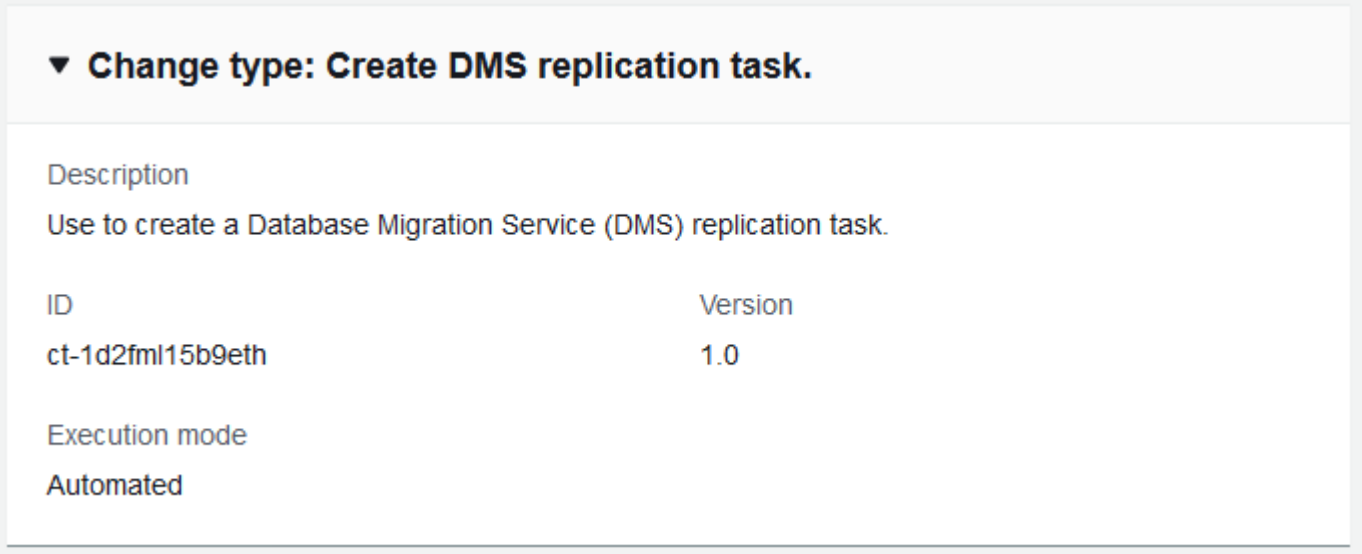
Change type ID	ct-1d2fml15b9eth
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create AWS DMS replication task

Creating a AWS DMS Replication Task with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a AWS DMS Replication Task with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1d2fml15b9eth" --change-type-version "1.0" --title "TestDMSRepTask" --
execution-parameters "{\"Description\":\"TestRepTask\",\"VpcId\":\"VPC-ID\",\"Name
\":\"DMSRepTask\",\"Parameters\":{\"CdcStartTime\":\1533776569\"MigrationType\":
\"full-load\",\"ReplicationInstanceArn\":\"REP_INSTANCE_ARN\",\"SourceEndpointArn
\":\"SOURCE_ENDPOINT_ARN\",\"TableMappings\":\"{\\\"rules\\\": [{\\\"rule-type
\\\": \\\"selection\\\",\\\"rule-id\\\": \\\"1\\\",\\\"rule-name\\\": \\\"1\\
\",\\\"object-locator\\\": {\\\"schema-name\\\": \\\"Test\\\",\\\"table-name\\
\": \\\"%\\\"}, \\\"rule-action\\\": \\\"include\\\"}] }\",\"TargetEndpointArn
\":\"TARGET_ENDPOINT_ARN\"},\"StackTemplateId\":\"stm-eos7uq0usnmeggdet\",
\"TimeoutInMinutes\":60}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateDmsRtParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1d2fml15b9eth" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsRtParams.json
```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
"Description":          "DMSTestRepTask",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-eos7uq0usnmeggdet",
"Name":                 "Test DMS RT",
"TimeoutInMinutes":     60,
"Parameters":   {
    "CdcStartTime":           "1533776569",
    "MigrationType":          "full-load",
    "ReplicationInstanceArn": "REP_INSTANCE_ARN",
    "SourceEndpointArn":      "SOURCE_ENDPOINT_ARN",
    "TargetEndpointArn":      "TARGET_ENDPOINT_ARN"
    "TableMappings":          {"rules": [{"rule-type": "selection","rule-id":
 "1","rule-name": "1","object-locator": {"schema-name": "Test","table-name": "%"},
 "rule-action": "include"}] }",
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsRtRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsRtRfc.json
```

4. Modify and save the CreateDmsRtRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1d2fml15b9eth",
"Title":                "DMS-RI-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsRtRfc.json --execution-
parameters file://CreateDmsRtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

You can create a AWS DMS task that captures three different types of changes or data. For more information, see Working with AWS DMS Tasks, Creating a Task, and Creating Tasks for Ongoing Replication Using AWS DMS.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1d2fml15b9eth.

Example: Required Parameters

```
{
  "Description": "Test Description",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "dmstask01",
  "StackTemplateId": "stm-eos7uq0usnmeggdet",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "MigrationType": "full-load",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789123:rep:6LDPDHOHCSDDSWHJP0UQJPAZLW",
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789123:endpoint:GLYHN8SUXLO5DS4PU4ODJ7KP0O",
    "TableMappings": "{     \"rules\": [         {             \"rule-type\":
 \"selection\",             \"rule-id\": \"1\",             \"rule-name\": \"1\",
       \"object-locator\": {                 \"schema-name\": \"Test\",
   \"table-name\": \"%\"             },             \"rule-action\": \"include\"
  }     ] }",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789123:endpoint:6PLJC4V60JGPKXN6OUOFWNJIUE"
  }
}
```

Example: All Parameters

```
{
  "Description": "Test Description",
  "VpcId": "vpc-317a9856",
  "Name": "dmstask01",
  "StackTemplateId": "stm-eos7uq0usnmeggdet",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CdcStartTime": "1533776569",
    "MigrationType": "full-load",
    "ReplicationInstanceArn": "arn:aws:dms:us-
east-1:123456789123:rep:6LDPDHOHCSDDSWHJP0UQJPAZLWTHISISVERYLONGANDIAMOKWITHIT",
    "ReplicationTaskIdentifier": "mydmstask01",
    "ReplicationTaskSettings": " {   \"TargetMetadata\": {     \"TargetSchema\": \"\",
     \"SupportLobs\": true,     \"FullLobMode\": false,     \"LobChunkSize\": 64,
     \"LimitedSizeLobMode\": true,     \"LobMaxSize\": 32,     \"BatchApplyEnabled
\": true   },   \"FullLoadSettings\": {     \"TargetTablePrepMode\": \"DO_NOTHING
\",     \"CreatePkAfterFullLoad\": false,     \"StopTaskCachedChangesApplied\":
 false,     \"StopTaskCachedChangesNotApplied\": false,     \"MaxFullLoadSubTasks
\": 8,     \"TransactionConsistencyTimeout\": 600,     \"CommitRate\": 10000   },
 \"Logging\": {     \"EnableLogging\": false   },   \"ControlTablesSettings\": {
  \"ControlSchema\":\"\",     \"HistoryTimeslotInMinutes\":5,     \"HistoryTableEnabled
\": false,     \"SuspendedTablesTableEnabled\": false,     \"StatusTableEnabled
\": false   },   \"StreamBufferSettings\": {     \"StreamBufferCount\": 3,
     \"StreamBufferSizeInMB\": 8   },   \"ChangeProcessingTuning\": {
  \"BatchApplyPreserveTransaction\": true,      \"BatchApplyTimeoutMin\": 1,
      \"BatchApplyTimeoutMax\": 30,      \"BatchApplyMemoryLimit\": 500,
 \"BatchSplitSize\": 0,      \"MinTransactionSize\": 1000,      \"CommitTimeout
\": 1,      \"MemoryLimitTotal\": 1024,      \"MemoryKeepTime\": 60,
 \"StatementCacheSize\": 50    },   \"ChangeProcessingDdlHandlingPolicy\": {
  \"HandleSourceTableDropped\": true,     \"HandleSourceTableTruncated\": true,
     \"HandleSourceTableAltered\": true   },   \"ValidationSettings\": {
  \"EnableValidation\": false,      \"ThreadCount\": 5   },   \"ErrorBehavior\": {
  \"DataErrorPolicy\": \"LOG_ERROR\",     \"DataTruncationErrorPolicy\":\"LOG_ERROR\",
     \"DataErrorEscalationPolicy\":\"SUSPEND_TABLE\",     \"DataErrorEscalationCount
\": 50,     \"TableErrorPolicy\":\"SUSPEND_TABLE\",     \"TableErrorEscalationPolicy
\":\"STOP_TASK\",     \"TableErrorEscalationCount\": 50,     \"RecoverableErrorCount
\": 0,     \"RecoverableErrorInterval\": 5,     \"RecoverableErrorThrottling\":
 true,     \"RecoverableErrorThrottlingMax\": 1800,     \"ApplyErrorDeletePolicy
\":\"IGNORE_RECORD\",     \"ApplyErrorInsertPolicy\":\"LOG_ERROR\",
 \"ApplyErrorUpdatePolicy\":\"LOG_ERROR\",     \"ApplyErrorEscalationPolicy\":
\"LOG_ERROR\",     \"ApplyErrorEscalationCount\": 0,     \"FullLoadIgnoreConflicts\":
 true   } }",
```

```
    "SourceEndpointArn": "arn:aws:dms:us-
east-1:123456789123:endpoint:GLYHN8SUXLO5DS4PU4ODJ7KP0OLONGSTRINGBUTITSOK",
    "TableMappings": "{     \"rules\": [         {             \"rule-type\":
 \"selection\",             \"rule-id\": \"1\",             \"rule-name\": \"1\",
        \"object-locator\": {                 \"schema-name\": \"Test\",
   \"table-name\": \"%\"             },             \"rule-action\": \"include\"
  }     ] }",
    "TargetEndpointArn": "arn:aws:dms:us-
east-1:123456789123:endpoint:6PLJC4V60JGPKXN6OUOFWNJIUEYOUSTILLDONTBELIEVEICANHANDLEVERYLONGSTR
  }
}
```


Database Migration Service (DMS) | Create Source Endpoint

Use to create a Database Migration Service (DMS) source endpoint.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint

Change Type Details

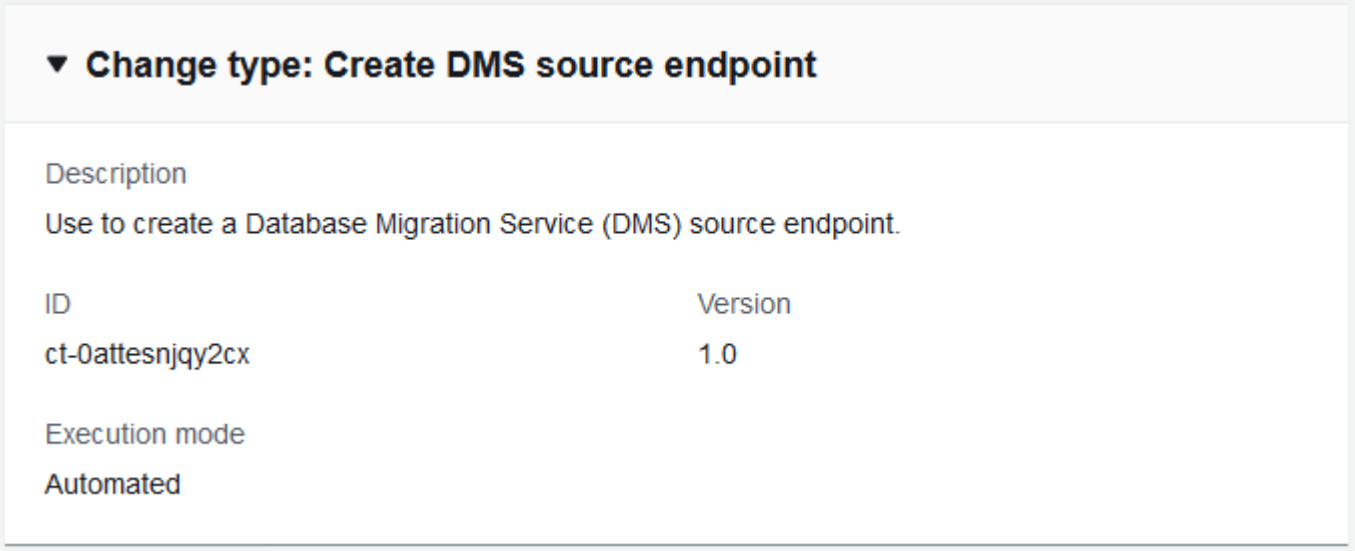
Change type ID	ct-0attesnjqy2cx
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

DMS source endpoint: creating

Creating a DMS Source Endpoint with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DMS Source Endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "MariaDB-DMS-
Source-Endpoint" --aws-account-id ACCOUNT-ID --change-type-id ct-0attesnjqy2cx --
change-type-version 1.0 --execution-parameters "{\"Description\":\"DESCRIPTION.\",
\"VpcId\":\"VPC-ID\",\"Name\":\"MariaDB-DMS-SE\",\"Parameters\":{\"EngineName\":
\"mariadb\",\"ServerName\":\"mariadb.db.example.com\",\"Port\":3306,\"Username\":
\"DB-USER\",\"Password\":\"DB-PW\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-
pud4ghhkp7395n9bc\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDmsSeParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0attesnjqy2cx" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "MariaDB-DMS-SE",
   "VpcId":                "VPC_ID",
   "Name":                 "Test SE",
   "StackTemplateId":      "stm-pud4ghhkp7395n9bc",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "Description":      "DESCRIPTION",
       "EngineName":       "mariadb",
       "ServerName":       "mariadb.db.example.com",
       "Port":             "3306",
       "Username":         "DB-USER",
       "Password":         "DB-PW",}
       }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsSeRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeRfc.json
```

4. Modify and save the CreateDmsSeRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0attesnjqy2cx",
"Title":                "MariaDB-DMS-Source-Endpoint"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsSeRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeRfc.json --execution-
parameters file://CreateDmsSeParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Before you create the DMS endpoint, make sure that your password doesn't contain unsupported characters. For more information, see Creating source and target endpoints in the *AWS Database Migration Service User Guide*.

To learn more, see Sources for Data Migration.

For an S3 source endpoint, see DMS source endpoint for S3: creating.

For a Mongo DB source endpoint, see DMS source endpoint for MongoDB: Creating.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0attesnjqy2cx.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "CertificateArn": "arn:aws:dms:us-
east-1:123456789121:cert:5957UBG4LS4ZJP2PK7YRYET6YE",
    "DatabaseName": "my-database",
    "EndpointIdentifier": "my-endpoint",
    "EngineName": "aurora",
    "KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Password": "$tr0n9PA55w0Rd",
    "Port": 50000,
    "ServerName": "",
    "SslMode": "none",
    "Username": ""
  },
  "StackTemplateId": "stm-pud4ghhkp7395n9bc",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

Database Migration Service (DMS) | Create Source Endpoint (MongoDB)

Use to create a Database Migration Service (DMS) source endpoint for MongoDB.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint (MongoDB)

Change Type Details

Change type ID	ct-2hxcllf1b4ey0
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required

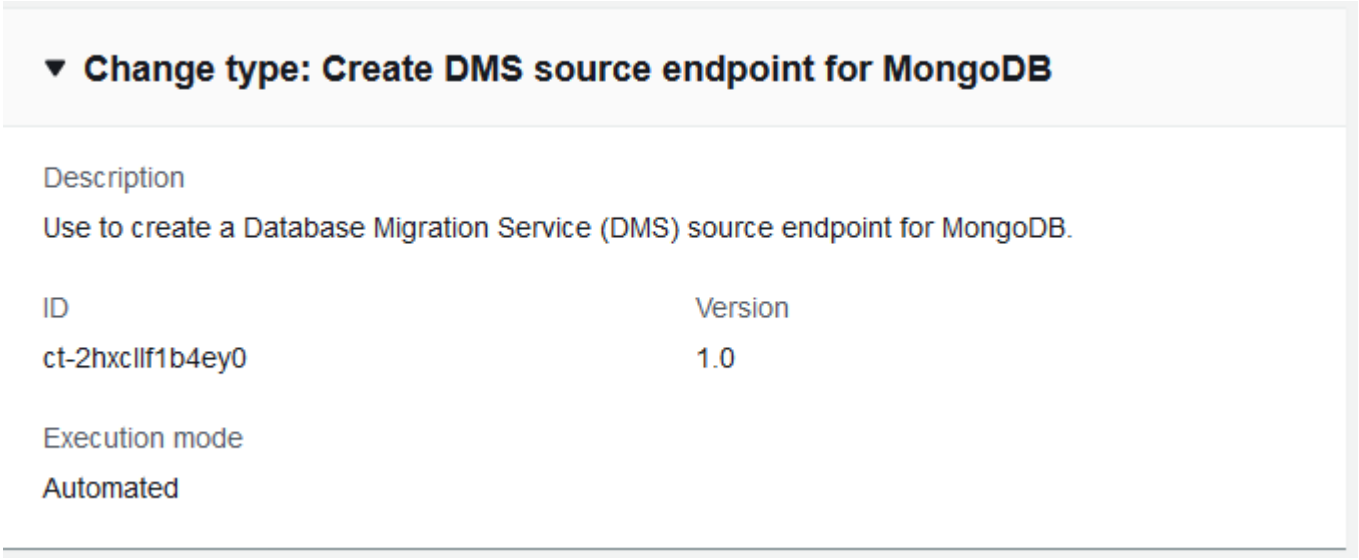
Execution mode	Automated

Additional Information

DMS source endpoint for MongoDB: Creating

Creating a DMS Mongo DB Source Endpoint with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DMS Mongo DB Source Endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-2hxcllf1b4ey0" --change-type-version "1.0" --title 'DMS_Source_MongoDB'
 --description "DESCRIPTION"  --execution-parameters "{\"Description\":
\"DMS_MongoDB_Source_Endpoint\",\"VpcId\":\"VPC_ID\",\"Name\":\"DMS-Mongo-SE\",
\"StackTemplateId\":\"stm-pud4ghhkp7395n9bc\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"DatabaseName\":\"mytestdb\",\"EngineName\":\"mongodb\",\"Port\":27017,\"ServerName
\":\"test.example.com\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDmsSeMongoParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hxcllf1b4ey0"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateDmsSeMongoParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "MongoDB-DMS-SE",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-pud4ghhkp7395n9bc",
   "Name":                 "Test Mongo SE",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "Description":      "DESCRIPTION",
       "DatabaseName":       "mytestdb",
   ```

```
    "EngineName":       "mongodb",
    "ServerName":       "test.example.com",
    "Port":             "27017"
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsSeMongoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeMongoRfc.json
```

4. Modify and save the CreateDmsSeMongoRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2hxcllf1b4ey0",
"Title":                "DMS_Source_MongoDB"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsSeMongoRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeMongoRfc.json --execution-
parameters file://CreateDmsSeMongoParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

AMS DMS can use Mongo or any Relational Database Service (RDS) as a source endpoint. For an S3 source endpoint, see DMS source endpoint for S3: creating.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2hxcllf1b4ey0.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "CertificateArn": "arn:aws:dms:us-
east-1:123456789121:cert:5957UBG4LS4ZJP2PK7YRYET6YE",
    "DatabaseName": "my-database",
    "EndpointIdentifier": "my-endpoint",
    "EngineName": "mongodb",
    "MongoDbAuthMechanism": "default",
    "MongoDbAuthSource": "admin",
    "MongoDbAuthType": "no",
    "MongoDbDocsToInvestigate": "1000",
    "MongoDbExtractDocId": "false",
    "MongoDbMetadataMode": "none",
    "Password": "$tr0n9PA55w0Rd",
    "Port": 27017,
    "ServerName": "my-server",
    "SslMode": "none",
    "Username": "my-user"
  },
  "StackTemplateId": "stm-pud4ghhkp7395n9bc",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

Database Migration Service (DMS) | Create Source Endpoint (S3)

Use to create a Database Migration Service (DMS) source endpoint for S3.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint (S3)

Change Type Details

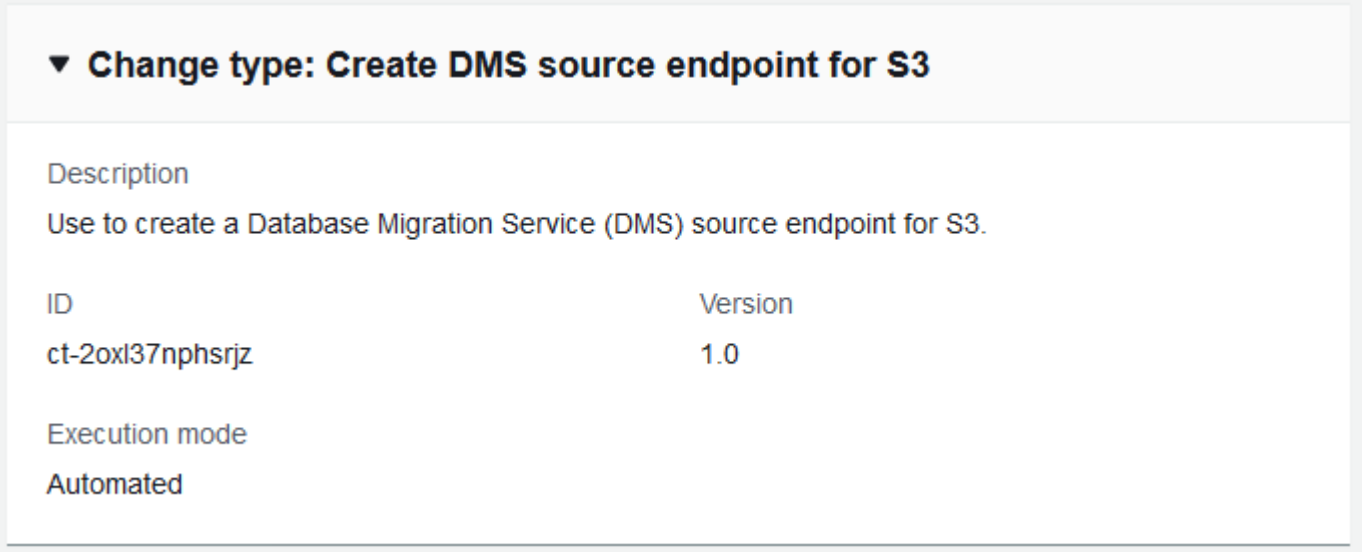
Change type ID	ct-2oxl37nphsrjz
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

DMS source endpoint for S3: creating

Creating a DMS S3 Source Endpoint with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DMS S3 Source Endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --title "S3DMSSourceEndpoint"
 --aws-account-id ACCOUNT-ID --change-type-id ct-2oxl37nphsrjz --change-type-version
 1.0 --execution-parameters "{\"Description\":\"TestS3DMS-SE\",\"VpcId\":\"VPC-ID\",
\"Name\":\"S3-DMS-SE\",\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"BUCKET-
NAME\",\"S3ExternalTableDefinition\":\"{\\\"TableCount\\\":\\\"1\\\",\\\"Tables\\\":[{\
\\"TableName\\\":\\\"employee\\\",\\\"TablePath\\\":\\\"hr/employee/\\\",\\\"TableOwner
\\\":\\\"hr\\\",\\\"TableColumns\\\":[{\\\"ColumnName\\\":\\\"Id\\\",\\\"ColumnType\\
\":\\\"INT8\\\",\\\"ColumnNullable\\\":\\\"false\\\",\\\"ColumnIsPk\\\":\\\"true\\\"},
{\\\"ColumnName\\\":\\\"LastName\\\",\\\"ColumnType\\\":\\\"STRING\\\",\\\"ColumnLength
\\\":\\\"20\\\"},{\\\"ColumnName\\\":\\\"FirstName\\\",\\\"ColumnType\\\":\\\"STRING\\
\",\\\"ColumnLength\\\":\\\"30\\\"},{\\\"ColumnName\\\":\\\"HireDate\\\",\\\"ColumnType
\\\":\\\"DATETIME\\\"},{\\\"ColumnName\\\":\\\"OfficeLocation\\\",\\\"ColumnType\\
\":\\\"STRING\\\",\\\"ColumnLength\\\":\\\"20\\\"}],\\\"TableColumnsTotal\\\":\\\"5\\
\"}]}\",\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789101:role/ams-ops-ct-authors-
dms-s3-test-role\"},\"TimeoutInMinutes\":60,\"StackTemplateId\":\"stm-pud4ghhkp7395n9bc
\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDmsSeS3Params.json.

```
aws amscm get-change-type-version --change-type-id "ct-2oxl37nphsrjz" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsSeS3Params.json
```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
"Description":          "TestS3DMS-SE",
"VpcId":                "VPC_ID",
"Name":                 "S3-DMS-SE",
"StackTemplateId":      "stm-pud4ghhkp7395n9bc",
"TimeoutInMinutes":     60,
"Parameters":   {
    "EngineName":               "s3",
    "S3BucketName":              "BUCKET-NAME",
    "S3ExternalTableDefinition":  "BUCKET-NAME",
    {"TableCount":                "1",
      "Tables":[{"TableName":"employee","TablePath":"hr/
employee/","TableOwner":"hr","TableColumns":
[{"ColumnName":"Id","ColumnType":"INT8","ColumnNullable":"false","ColumnIsPk":"true"},
{"ColumnName":"LastName","ColumnType":"STRING","ColumnLength":"20"},
{"ColumnName":"FirstName","ColumnType":"STRING","ColumnLength":"30"},
{"ColumnName":"HireDate","ColumnType":"DATETIME"},
{"ColumnName":"OfficeLocation","ColumnType":"STRING","ColumnLength":"20"}],"TableColumnsTot
    "S3ServiceAccessRoleArn":      "arn:aws:iam::123456789101:role/ams-ops-ct-
authors-dms-s3-test-role",
      }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsSeS3Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsSeS3Rfc.json
```

4. Modify and save the CreateDmsSeS3Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
```

```
"ChangeTypeId":          "ct-2oxl37nphsrjz",
"Title":                 "DMS_Source_S3"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsSeS3Rfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsSeS3Rfc.json --execution-
parameters file://CreateDmsSeS3Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

AMS DMS can use S3 or any Relational Database Service (RDS) source endpoint. For a Mongo DB source endpoint, see DMS source endpoint for MongoDB: Creating.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2oxl37nphsrjz.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "EndpointIdentifier": "my-endpoint",
    "EngineName": "s3",
```

```
    "S3BucketFolder": "my-folder",
    "S3BucketName": "my-bucket",
    "S3CompressionType": "NONE",
    "S3CsvDelimiter": ",",
    "S3CsvRowDelimiter": "\n",
    "S3ExternalTableDefinition": "false",
    "S3ServiceAccessRoleArn": "arn:aws:iam::123456789012:role/my-s3service-role"
  },
  "StackTemplateId": "stm-pud4ghhkp7395n9bc",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

Database Migration Service (DMS) | Create Target Endpoint

Use to create a Database Migration Service (DMS) target endpoint for RDS supported MySQL, MariaDB, PostgreSQL, Oracle and Microsoft SQL server engine.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create target endpoint

Change Type Details

Change type ID	ct-3gf8dolbo8x9p
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

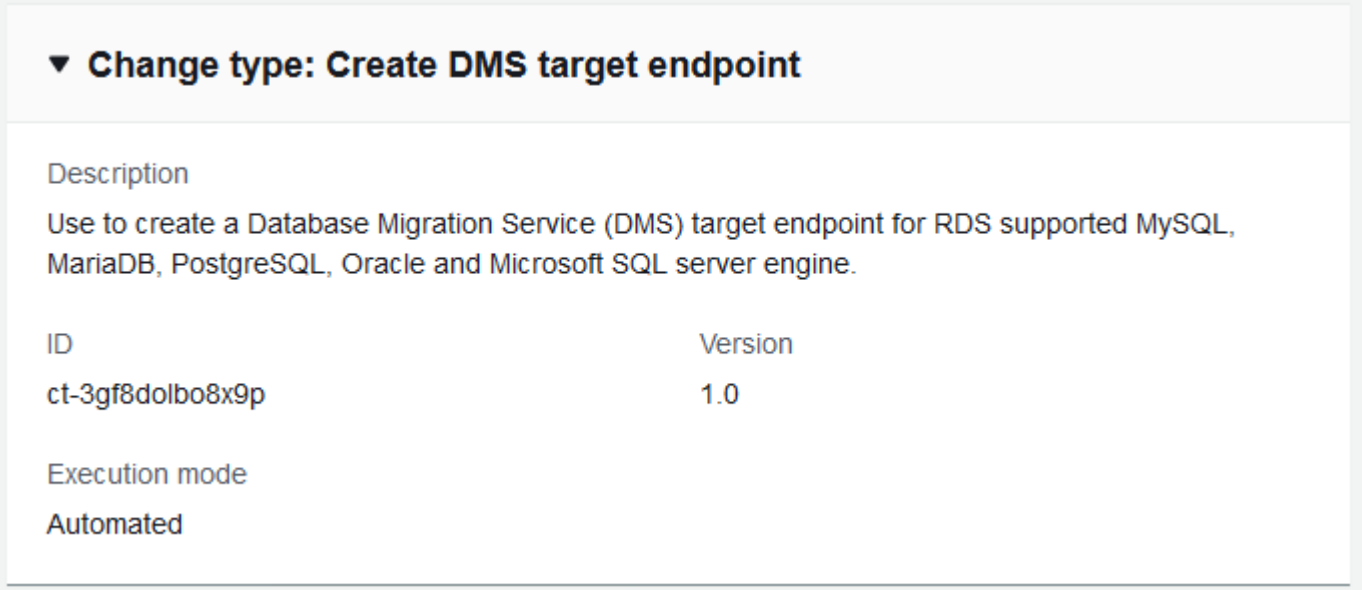
Additional Information

DMS target endpoint: creating

AMS DMS can use S3 or any Relational Database Service (RDS) with MySQL, MariaDB, Oracle, Postgresql, or Microsoft SQL as a target endpoint.

Creating a DMS Target Endpoint with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DMS Target Endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3gf8dolbo8x9p" --change-type-version "1.0" --title "TestDMSTargetEndpointSql" --
execution-parameters "{\"Description\":\"TestSQLTE\",\"VpcId\":\"VPC-ID\",\"Name\":
\"SQLTE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes\":60,
\"Parameters\":{\"EngineName\":\"mysql\",\"Password\":\"testpw123\",\"Port\":\"3306\",
\"ServerName\":\"mytestdb.d5fga0rf2wpi.ap-southeast-2.rds.amazonaws.com\",\"Username\":
\"USERNAME\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDmsTeParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3gf8dolbo8x9p" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "TestSQLTE",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-knghtmmgefafdq89u",
   "Name":                 "SQLTE-NAME",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "EngineName":       "mysql",
       "ServerName":       "sql.db.example.com",
       "Port":             "3306",
       "Username":         "DB-USER",
       "Password":         "DB-PW",}
       }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsTeRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeRfc.json
```

4. Modify and save the CreateDmsTeRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3gf8dolbo8x9p",
"Title":                "SQLDB-DMS-Target-Endpoint"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsTeRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeRfc.json --execution-
parameters file://CreateDmsTeParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

AMS DMS can use S3 or any Relational Database Service (RDS) with MySQL, MariaDB, Oracle, Postgresql, or Microsoft SQL as a target endpoint. For an S3 target endpoint, see DMS target endpoint for S3: creating.

For more information, see Targets for Data Migration.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3gf8dolbo8x9p.

Example: Required Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "dmstarget01",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "StackTemplateId": "stm-knghtmmgefafdq89u",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "EngineName": "mysql",
    "Password": "testpasswrod123",
    "Port": "3306",
    "ServerName": "mytestdb.d5fga0rf2wpi.ap-southeast-2.rds.amazonaws.com",
    "Username": "myuser01"
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-104ed2fb",
  "Name": "dmstarget01",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
```

```
  "StackTemplateId": "stm-knghtmmgefafdq89u",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CertificateArn": "arn:aws:dms:us-
east-1:123456789121:cert:5957UBG4LS4ZJP2PK7YRYET6YE",
    "DatabaseName": "mytestdb",
    "EndpointIdentifier": "myctdmstarget",
    "EngineName": "mysql",
    "ExtraConnectionAttributes": "targetDbType=MULTIPLE_DATABASES",
    "KmsKeyId": "15a25b6b-b29d-4bc5-af34-88eeb3740a94",
    "Password": "testpasswrod123",
    "Port": "3306",
    "ServerName": "mytestdb.d5fga0rf2wpi.ap-southeast-2.rds.amazonaws.com",
    "SslMode": "verify-full",
    "Username": "myuser01"
  }
}
```

Database Migration Service (DMS) | Create Target Endpoint (S3)

Use to create a Database Migration Service (DMS) target endpoint for S3.

Full classification: Deployment | Advanced stack components | Database Migration Service (DMS) | Create target endpoint (S3)

Change Type Details

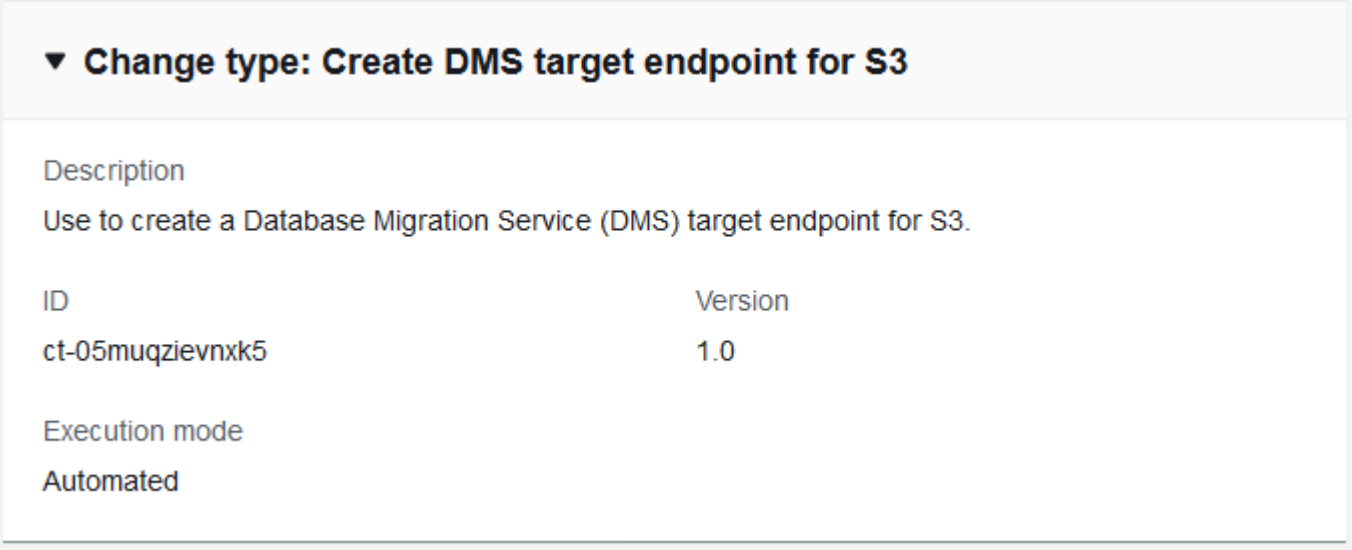
Change type ID	ct-05muqzievnxk5
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

DMS target endpoint for S3: creating

Creating a DMS S3 Target Endpoint with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DMS S3 Target Endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-05muqzievnxk5" --change-type-version "1.0" --title "TestDMSTargetEndpointS3"
 --execution-parameters "{\"Description\":\"TestS3TE\",\"VpcId\":\"VPC-ID\",\"Name
\":\"S3TE-NAME\",\"StackTemplateId\":\"stm-knghtmmgefafdq89u\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"EngineName\":\"s3\",\"S3BucketName\":\"mybucket.in.s3\",
\"S3ServiceAccessRoleArn\":\"arn:aws:iam::123456789123:role/my-s3-role\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateDmsTeS3Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-05muqzievnxk5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateDmsTeS3Params.json
   ```

2. Modify and save the execution parameters CreateDmsTeS3Params.json file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "TestS3DMS-TE",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-knghtmmgefafdq89u",
   "Name":                 "DMS-S3-TE",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "EngineName":       "s3",
       "S3BucketName":      "BUCKET-NAME",
       "S3ServiceAccessRoleArn":       "arn:aws:iam::123456789101:role/ams-ops-ct-
   authors-dms-s3-test-role"
         }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateDmsTeS3Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDmsTeS3Rfc.json
```

4. Modify and save the CreateDmsTeS3Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-05muqzievnxk5",
"Title":                "DMS_Target_S3"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDmsTeS3Rfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDmsTeS3Rfc.json --execution-
parameters file://CreateDmsTeS3Params.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

AMS provides a separate change type for creating a target endpoint for S3. For more information, see Using Amazon S3 as a Target for AWS Database Migration Service and Extra Connection Attributes When Using Amazon S3 as a Target for AWS DMS.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-05muqzievnxk5.

Example: Required Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "dmstargets301",
  "StackTemplateId": "stm-knghtmmgefafdq89u",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "EngineName": "s3",
    "S3BucketName": "mybucket.in.s3",
    "S3ServiceAccessRoleArn": "arn:aws:iam::123456789123:role/my-s3-role"
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-317a9856",
  "Name": "dmstargets301",
  "StackTemplateId": "stm-knghtmmgefafdq89u",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "EndpointIdentifier": "mys3dmstarget",
    "EngineName": "s3",
    "ExtraConnectionAttributes": "maxFileSize=512",
    "S3BucketFolder": "mytestfolder",
    "S3BucketName": "mybucket.in.s3",
    "S3CompressionType": "NONE",
    "S3CsvDelimiter": "|",
    "S3CsvRowDelimiter": "M",
    "S3ServiceAccessRoleArn": "arn:aws:iam::123456789123:role/my-s3-role"
  }
}
```

DNS (Private) | Create

Create a new Route 53 DNS resource record sets and a new private hosted zone for a VPC, and configure traffic routing.

Full classification: Deployment | Advanced stack components | DNS (private) | Create

Change Type Details

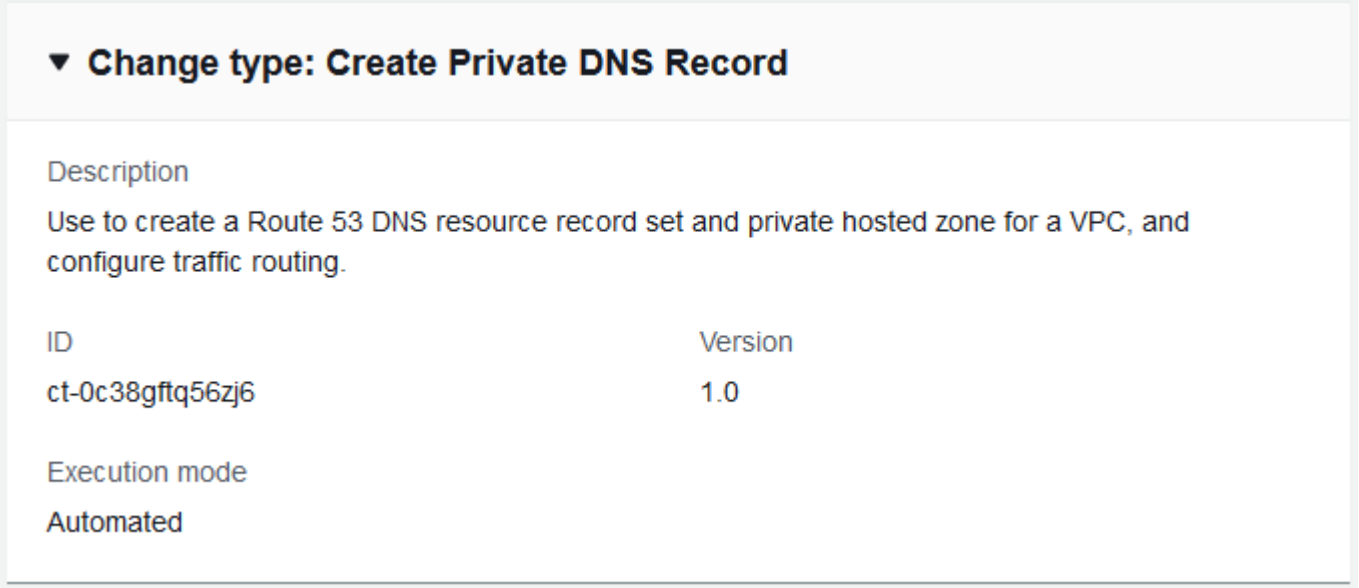
Change type ID	ct-0c38gftq56zj6
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create private DNS Route 53

Creating a private Route 53 hosted zone with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a private Route 53 hosted zone with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-0c38gftq56zj6" \
--change-type-version "2.0" --title "Testing - Creating New Private Hosted Zone" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
CreateAddRoute53Resources\",\"Region\":\"us-east-1\",\"Parameters\":{\"DomainName\":
\"mydomain.com\",\"VPCId\":\"vpc-12345678\",\"DomainType\":\"private\",\"RecordSet\":
[\"[{\\\"Name\\\":\\\"test1.mydomain.com\\\",\\\"Type\\\":\\\"A\\\",\\\"TTL\\\":600,\\
\"ResourceRecords\\\":[\\\"10.1.1.1\\\",\\\"10.1.2.2\\\"]}]}\"]}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDnsPrivateParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0c38gftq56zj6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateDnsPrivateParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
    "DocumentName": "AWSManagedServices-CreateAddRoute53Resources",
    "Region": "us-east-1",
    "Parameters": {
        "DomainName": "mydomain.com",
        "VpcId": "vpc-12345678",
        "DomainType": "private",
        "RecordSet": [
            "{\"RecordSet\":[{\"Name\":\"test1.mydomain.com\",\"Type\":\"A\",\"TTL
\":600,\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]}]}"
        ]
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDnsPrivateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDnsPrivateRfc.json
```

4. Modify and save the CreateDnsPrivateRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-0c38gftq56zj6",
  "ChangeTypeVersion": "2.0",
  "Title": "Creating New Private Hosted Zone"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDnsPrivateRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDnsPrivateRfc.json --execution-
parameters file://CreateDnsPrivateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- This CT fails if the specified **RecordSet** contains more than 500 resource records (RRs), or if the CloudFormation template surpasses the maximum body of 51,200 bytes.
- To create a public Route 53 DNS stack, see Create public DNS Route 53.

  To update an existing private Route 53 DNS stack, see Update private DNS Route 53.
- For **RecordSetType** = A, be sure to specify either **AliasTargetDnsName**or **RecordSetValue**.
- You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.
- For more information, see Working with Private Hosted Zones.

  To update your private DNS stack after it's created, see Update private DNS Route 53.

  To create a public Route 53 DNS stack, see Create public DNS Route 53.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0c38gftq56zj6.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAddRoute53Resources",
  "Region" : "ap-southeast-2",
  "Parameters": {
    "DomainName": "mydomain.com",
    "VPCId": "vpc-5a25bd3f",
    "DomainType": "private",
    "RecordSet": [
      "{\"RecordSet\":[{\"Name\":\"test1.mydomain.com\",\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test3.mydomain.com
\",\"Type\":\"CNAME\",\"TTL\":\"600\",\"ResourceRecords\":[\"amazon.com\"]},
{\"Name\":\"test4.mydomain.com\",\"Type\":\"A\",\"AliasTarget\":{\"DNSName\":
\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,\"HostedZoneId
\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"weighted.mydomain.com\",\"Weight\":200,
```

```
\"SetIdentifier\":\"Example-Set-Identifier-1\",\"Type\":\"A\",\"AliasTarget\":
{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"geolocationexample.mydomain.com
\",\"SetIdentifier\":\"Example-GeoLocation-Identifier-1\",\"GeoLocation\":
{\"CountryCode\":\"US\",\"SubdivisionCode\":\"WA\"},\"Type\":\"A\",\"AliasTarget
\":{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"examplelatency.mydomain.com\",
\"SetIdentifier\":\"Example-Latency-Identifier-1\",\"Region\":\"ap-southeast-2\",
\"Type\":\"A\",\"TTL\":\"600\",\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},
{\"Name\":\"examplemultivalue.mydomain.com\",\"SetIdentifier\":\"Example-
MultiValue-Identifier-1\",\"MultiValueAnswer\":true,\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\"]}]}"
    ]
  }
}
```

DNS (Public) | Create

Create a new Route 53 DNS resource record set and a new public hosted zone for a VPC, and configure traffic routing.

Full classification: Deployment | Advanced stack components | DNS (public) | Create

Change Type Details

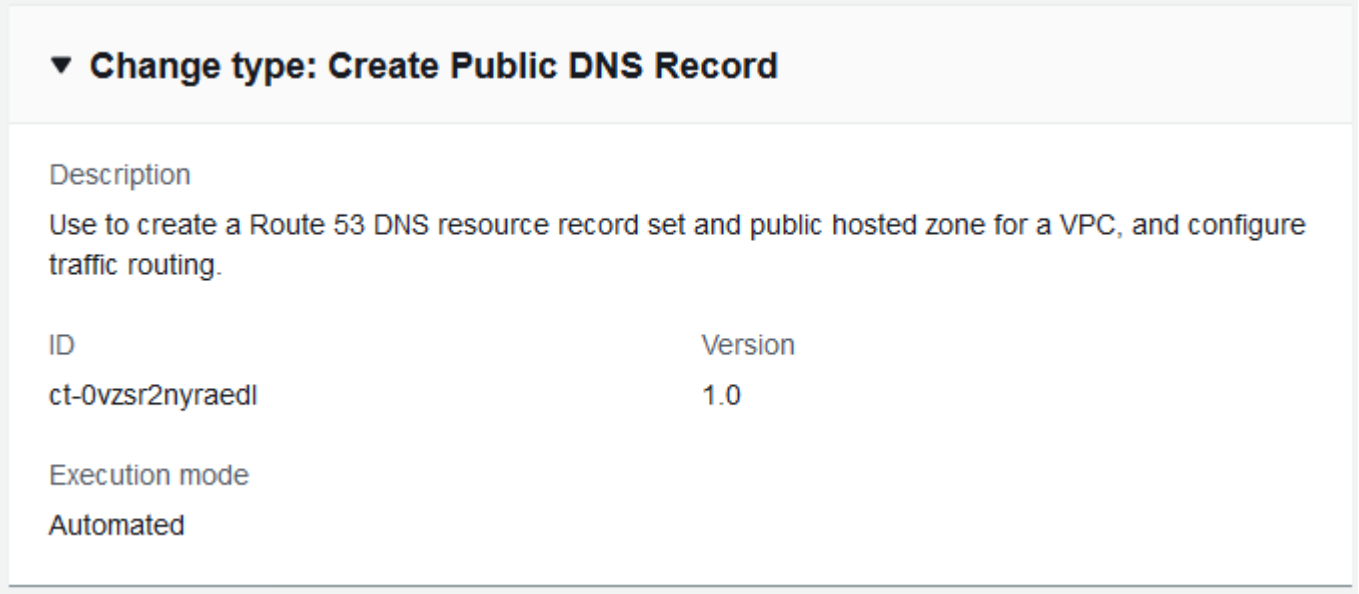
Change type ID	ct-0vzsr2nyraedl
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create public DNS Route 53

Creating a public Route 53 hosted zone with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a public Route 53 hosted zone with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-0vzsr2nyraedl" \
--change-type-version "2.0" --title "Creating New Public Hosted Zone" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
CreateAddRoute53Resources\",\"Region\":\"us-east-1\",\"Parameters\":{\"DomainName
\":\"mydomain.com\",\"DomainType\":\"public\",\"RecordSet\":[\"[{\\\"Name\\\":\\
\"test1.mydomain.com\\\",\\\"Type\\\":\\\"A\\\",\\\"TTL\\\":600,\\\"ResourceRecords\\
\":[\\\"10.1.1.1\\\",\\\"10.1.2.2\\\"]}]}\"]}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateDnsPublicParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0vzsr2nyraedl"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateDnsPublicParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateAddRoute53Resources",
     "Region": "ap-southeast-2",
     "Parameters": {
       "DomainName": "domain.com",
       "DomainType": "public",
       "RecordSet": [
         "{\"RecordSet\":[{\"Name\":\"test1.domain.com\",\"Type\":\"A\",\"TTL\":600,
   \"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test3.domain.com\",
   \"Type\":\"CNAME\",\"TTL\":600,\"ResourceRecords\":[\"www.google.com\"]},
   ```

```
{\"Name\":\"test4.domain.com\",\"Type\":\"A\",\"AliasTarget\":{\"DNSName\":
\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,\"HostedZoneId
\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"weighted.domain.com\",\"Weight\":200,
\"SetIdentifier\":\"Example-Set-Identifier-1\",\"Type\":\"A\",\"AliasTarget\":
{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"geolocationexample.domain.com\",
\"SetIdentifier\":\"Example-GeoLocation-Identifier-1\",\"GeoLocation\":
{\"CountryCode\":\"US\",\"SubdivisionCode\":\"WA\"},\"Type\":\"A\",\"AliasTarget
\":{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"examplelatency.domain.com\",
\"SetIdentifier\":\"Example-Latency-Identifier-1\",\"Region\":\"ap-southeast-2\",
\"Type\":\"A\",\"TTL\":600,\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},
{\"Name\":\"examplemultivalue.domain.com\",\"SetIdentifier\":\"Example-
MultiValue-Identifier-1\",\"MultiValueAnswer\":true,\"Type\":\"A\",\"TTL\":600,
\"ResourceRecords\":[\"10.1.1.1\"]}]}"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateDnsPublicRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDnsPublicRfc.json
```

4. Modify and save the CreateDnsPublicRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-0vzsr2nyraedl",
"Title":                "DNS-Public-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateDnsPublicRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateDnsPublicRfc.json --execution-
parameters file://CreateDnsPublicParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> To create a private Route 53 DNS stack, see Create private DNS Route 53.

> **Note**
>
> For **RecordSetType** = A, be sure to specify either **AliasTargetDnsName**or **RecordSetValue**.

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more, see Working with Public Hosted Zones.

To update your public DNS stack after it's created, see Update public DNS Route 53.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0vzsr2nyraedl.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAddRoute53Resources",
  "Region" : "us-east-1",
  "Parameters": {
    "DomainName": "mydomain.com",
    "DomainType": "public",
```

```
    "RecordSet": [
      "{\"RecordSet\":[{\"Name\":\"test1.mydomain.com\",\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test3.mydomain.com
\",\"Type\":\"CNAME\",\"TTL\":\"600\",\"ResourceRecords\":[\"amazon.com\"]},
{\"Name\":\"test4.mydomain.com\",\"Type\":\"A\",\"AliasTarget\":{\"DNSName\":
\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,\"HostedZoneId
\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"weighted.mydomain.com\",\"Weight\":200,
\"SetIdentifier\":\"Example-Set-Identifier-1\",\"Type\":\"A\",\"AliasTarget\":
{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"geolocationexample.mydomain.com
\",\"SetIdentifier\":\"Example-GeoLocation-Identifier-1\",\"GeoLocation\":
{\"CountryCode\":\"US\",\"SubdivisionCode\":\"WA\"},\"Type\":\"A\",\"AliasTarget
\":{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"examplelatency.mydomain.com\",
\"SetIdentifier\":\"Example-Latency-Identifier-1\",\"Region\":\"ap-southeast-2\",
\"Type\":\"A\",\"TTL\":\"600\",\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},
{\"Name\":\"examplemultivalue.mydomain.com\",\"SetIdentifier\":\"Example-
MultiValue-Identifier-1\",\"MultiValueAnswer\":true,\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\"]}]}"
    ]
  }
}
```

DynamoDB | Create from Backup

Create an Amazon DynamoDB stack from backup.

Full classification: Deployment | Advanced stack components | DynamoDB | Create from backup

Change Type Details

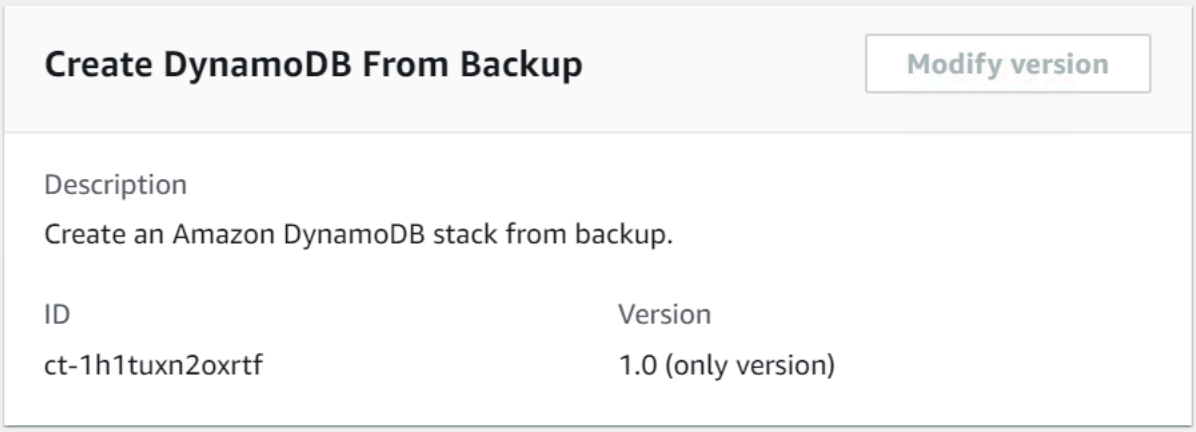
Change type ID	ct-1h1tuxn2oxrtf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

DynamoDB (creating from backup)

Creating a DynamoDb with an AWS Backup with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a DynamoDb with an AWS Backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1h1tuxn2oxrtf" \
--change-type-version "1.0" --title "AWS Backup Start Restore Job for DynamoDB" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-StartRestoreJobDynamoDB
\",\"Region\":\"us-east-1\",\"Parameters\":{\"BackupVaultName\":[\"Default\"],
\"RecoveryPointArn\":[\"arn:aws:dynamodb:us-east-1:000000000000:table/table-name/
backup/00000000000000-00000000\"],\"TargetTableName\":[\"TARGET_TABLE_NAME\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it RestoreJobRestoreDynamoDbParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1h1tuxn2oxrtf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RestoreJobRestoreDynamoDbParams.json
   ```

2. Modify and save the RestoreJobRestoreDynamoDbParams file.

   ```
   {
     "DocumentName": "AWSManagedServices-StartRestoreJobDynamoDB",
     "Region": "us-east-1",
     "Parameters": {
       "BackupVaultName": ["Default"],
       "RecoveryPointArn": ["arn:aws:dynamodb:us-east-1:000000000000:table/table-name/
   backup/00000000000000-00000000"],
       "TargetTableName": ["TARGET_TABLE_NAME"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it RestoreJobRestoreDynamoDbRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RestoreJobRestoreDynamoDbRfc.json
   ```

4. Modify and save the RestoreJobRestoreDynamoDbRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-1h1tuxn2oxrtf",
    "ChangeTypeVersion": "1.0",
    "Title": "AWS Backup Start Restore Job for DynamoDB"
}
```

5. Create the RFC, specifying the RestoreJobRestoreDynamoDbRfc file and the RestoreJobRestoreDynamoDbParams file:

```
aws amscm create-rfc --cli-input-json file://RestoreJobRestoreDynamoDbRfc.json --
execution-parameters file://RestoreJobRestoreDynamoDbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS, see Amazon DynamoDB.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1h1tuxn2oxrtf.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobDynamoDB",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["Vault01"],
    "RecoveryPointArn": ["arn:aws:dynamodb:us-east-1::table/xyz-test/
backup/01585118622000-75ee13f1"],
```

```
    "TargetTableName": ["New-TargetTable"]
  }
}
```

EBS Snapshot | Copy

Copy an Elastic Block Store (EBS) snapshot in your AMS account.

Full classification: Deployment | Advanced stack components | EBS snapshot | Copy

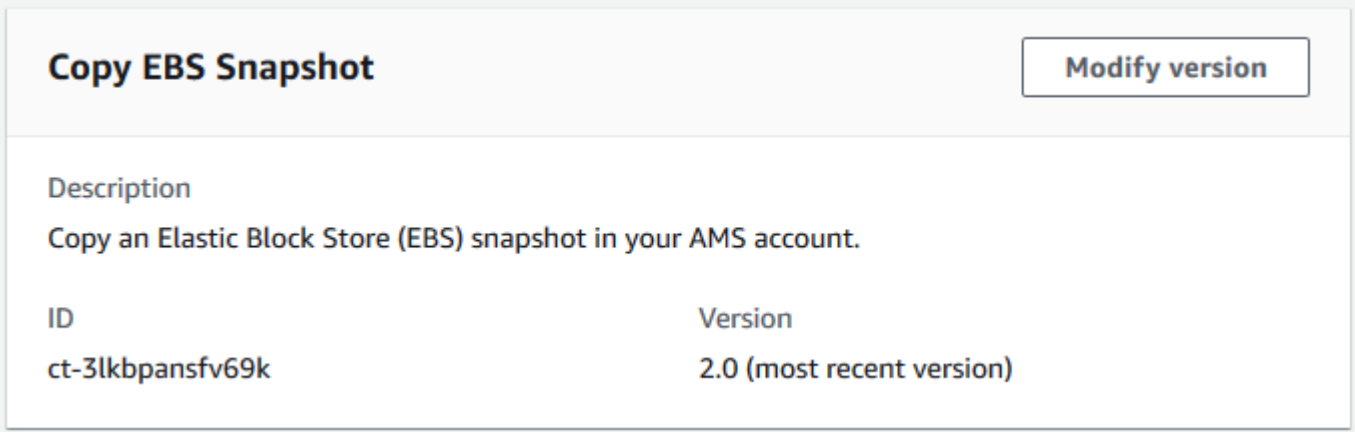
Change Type Details

Change type ID	ct-3lkbpansfv69k
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Copy EBS snapshot

Copying EBS Snapshots with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Copying EBS Snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3lkbpansfv69k" --change-type-version
 "2.0" --title "Copy EBS snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CopyEBSSnapshot\",\"Region\":\"us-east-1\",\"Parameters\":
{\"SourceSnapshotId\":[\"SNAPSHOT_ID\"],\"SourceRegion\":[\"ap-southeast-2\"],
\"KmsKeyId\":[\"KEY_ID\"],\"Description\":[\"test-snapshot\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CopyEbsSnpshtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3lkbpansfv69k" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CopyEbsSnpshtParams.json
   ```

2. Modify and save the CopyEbsSnpshtParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-CopyEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceSnapshotId": [
      "SNAPSHOT_ID"
    ],
    "SourceRegion": [
      "ap-southeast-2"
    ],
    "KmsKeyId": [
      "KEY_ID"
    ],
    "Description": [
      "test-snapshot"
    ]
  }
}
```

3. Output the RFC template JSON file to a file; this example names it CopyEbsSnpshtRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CopyEbsSnpshtRfc.json
```

4. Modify and save the CopyEbsSnpshtRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-3lkbpansfv69k",
  "Title": "Copy EBS snapshot"
}
```

5. Create the RFC, specifying the CopyEbsSnpshtRfc file and the CopyEbsSnpshtParams file:

```
aws amscm create-rfc --cli-input-json file://CopyEbsSnpshtRfc.json  --execution-
parameters file://CopyEbsSnpshtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> A typical use for the EBS snapshot share and copy CTs would be:
>
> 1. In account A, use the Share EBS snapshot CT to share the snapshot with account B.
> 2. In account B, use the EBS snapshot copy CT to copy the snapshot to the AWS Region for account B.

> **Important**
>
> This change type version, 2.0, removes several parameters, **TargetParameterName**, **Targets**, **MaxConcurrency**, and **MaxErrors**; and introduces one new parameter, **SourceSnapshotId**.

To learn more about Amazon EBS snapshots, see Amazon EBS Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3lkbpansfv69k.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceRegion": [
      "us-east-1"
    ],
    "SourceSnapshotId": [
      "snap-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceRegion": [
      "us-east-1"
    ],
    "SourceSnapshotId": [
      "snap-1234567890abcdef0"
    ],
    "KmsKeyId": [
      "01234567-abcd-abcd-abcd-0123456789ab"
    ],
    "Description": [
      "my-snapshot"
    ]
  }
}
```

EBS Snapshot | Create

Create an Elastic Block Store (EBS) snapshot from an EBS volume. The volume must be attached to an EC2 instance.

Full classification: Deployment | Advanced stack components | EBS snapshot | Create

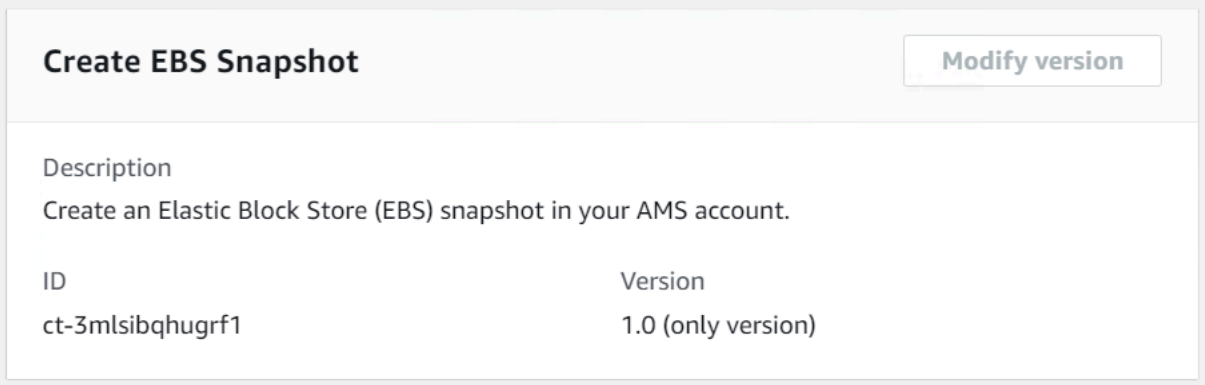
Change Type Details

Change type ID	ct-3mlsibqhugrf1
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create EBS snapshot

Creating EBS Snapshots with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating EBS Snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3mlsibqhugrf1" --change-type-version
 "1.0" --title "Create EBS snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CreateEBSSnapshot\",\"Region\":\"us-east-1\",\"Parameters\":
{\"VolumeId\":[\"VOL_ID\"],\"Description\":[\"My snapshot\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateEbsSnpshtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3mlsibqhugrf1"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateEbsSnpshtParams.json
   ```

2. Modify and save the CreateEbsSnpshtParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-CreateEBSSnapshot",
       "Region": "us-east-1",
       "Parameters": {
           "VolumeId": [
               "VOL_ID"
           ],
           "Description": [
               "My snapshot"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it CreateEbsSnpshtRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateEbsSnpshtRfc.json
   ```

4. Modify and save the CreateEbsSnpshtRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3mlsibqhugrf1",
  "Title": "Create EBS snapshot"
}
```

5. Create the RFC, specifying the CreateEbsSnpshtRfc file and the CreateEbsSnpshtParams file:

```
aws amscm create-rfc --cli-input-json file://CreateEbsSnpshtRfc.json  --execution-
parameters file://CreateEbsSnpshtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS snapshots, see Amazon EBS Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3mlsibqhugrf1.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeId": [
      "vol-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateEBSSnapshot",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "VolumeId": [
      "vol-1234567890abcdef0"
    ],
    "Description": [
      "my-snapshot"
    ]
  }
}
```

EBS Volume | Create

Creates up to five EBS volumes, and attaches them to an existing EC2 instance that you specify. Does not create a root volume.

Full classification: Deployment | Advanced stack components | EBS Volume | Create

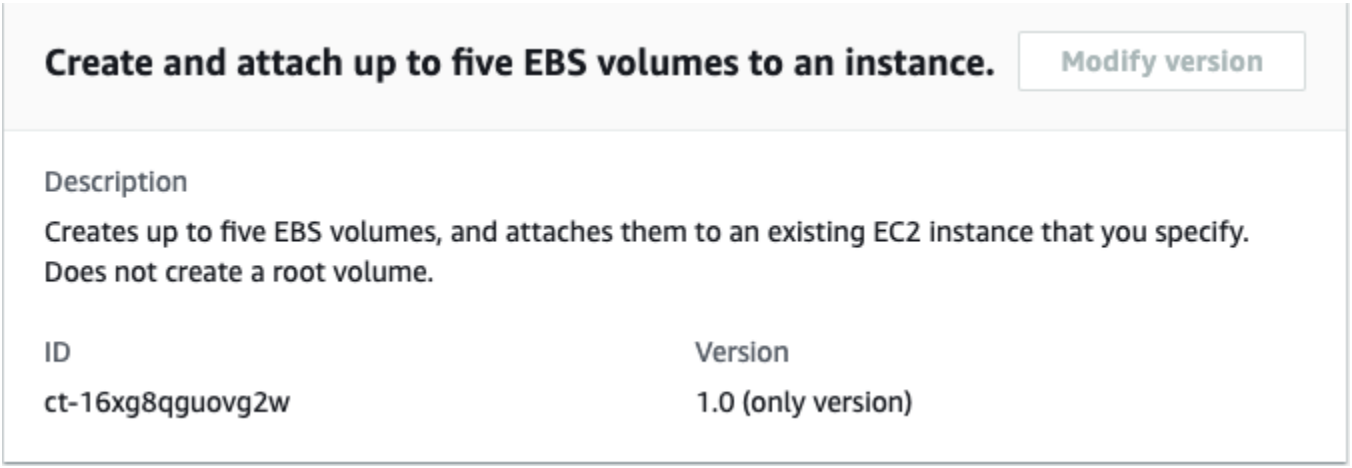
Change Type Details

Change type ID	ct-16xg8qguovg2w
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create EBS volume

Creating EBS Volumes with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating EBS Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-16xg8qguovg2w" --change-type-version "1.0"
 --title "EBS-Create-RFC" --execution-parameters "{\"Description\": \"Create 2 volumes
 and attach to i-12345678901234567\",\"VpcId\": \"vpc-0a60eb65b4EXAMPLE\",\"Name\":
 \"EBSVolumeStack\",\"StackTemplateId\": \"stm-hrnfpt7l0qqumcelt\",\"TimeoutInMinutes
\":\"45\",\"Parameters\": {\"AvailabilityZone\": \"us-east-1d\",\"InstanceId\":
 \"i-12345678901234567\",\"Volume1Name\": \"/dev/xvdf\",\"Volume1Size\": \"20\",
\"Volume1Type\": \"gp3\",\"Volume1Iops\": \"3000\",\"Volume1Throughput\": \"125\",
\"Volume2Name\": \"/dev/xvdg\",\"Volume2Size\": \"20\",\"Volume2Iops\": \"200\",
\"Volume2Type\": \"io2\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateEbsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-16xg8qguovg2w" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateEbsParams.json
   ```

2. Modify and save the CreateEbsParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Description":    "Create 2 volumes and attach to i-12345678901234567.",
     "VpcId":          "vpc-0a60eb65b4EXAMPLE",
     "Name":           "EBSVolumeStack",
     "StackTemplateId": "stm-hrnfpt7l0qqumcelt",
     "TimeoutInMinutes": "45",
     "Parameters": {
       "AvailabilityZone": "us-east-1a",
       "InstanceId":       "i-12345678901234567",
       "Volume1Name":      "/dev/xvdf",
       "Volume1Size":      "20",
       "Volume1Type":      "gp3"
       "Volume1Iops": "3000",
       "Volume1Throughput": "125",
       "Volume2Name": "/dev/xvdg",
   ```

```
    "Volume2Size": "20",
    "Volume2Iops": "200",
    "Volume2Type": "io2"
  }
}
```

3. Output the RFC template JSON file to a file; this example names it CreateEbsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateEbsRfc.json
```

4. Modify and save the CreateEbsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-16xg8qguovg2w",
"Title":                "EBS-Create-RFC"
}
```

5. Create the RFC, specifying the CreateEbsRfc file and the CreateEbsParams file:

```
aws amscm create-rfc --cli-input-json file://CreateEbsRfc.json  --execution-
parameters file://CreateEbsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS, see Amazon Elastic Block Store (EBS).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-16xg8qguovg2w.

Example: Required Parameters

```
{
    "Description": "This is a test description",
    "VpcId": "vpc-1234567890abcdef0",
```

```
    "Name": "Test Stack",
    "Parameters": {
      "AvailabilityZone": "us-west-2a",
      "InstanceId": "i-04ca00332c8e8c226",
      "Volume1Name": "/dev/xvdbb",
      "Volume1Size": "100",
      "Volume1Type": "gp2"
    },
    "TimeoutInMinutes": 60,
    "StackTemplateId": "stm-hrnfpt7l0qqumcelt"
  }
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "Parameters": {
    "AvailabilityZone": "us-west-2a",
    "InstanceId": "i-04ca00332c8e8c226",
    "Volume1Iops": "3000",
    "Volume1Throughput": "125",
    "Volume1KmsKeyId": "225bc21e-fc82-4388-8c91-855f3cadb63c",
    "Volume1Name": "/dev/xvdbb",
    "Volume1Size": "100",
    "Volume1Snapshot": "snap-12345678",
    "Volume1Type": "gp3",
    "Volume2Iops": "3000",
    "Volume2Throughput": "125",
    "Volume2KmsKeyId": "225bc21e-fc82-4388-8c91-855f3cadb63c",
    "Volume2Name": "/dev/xvdbc",
    "Volume2Size": "100",
```

```
    "Volume2Snapshot": "snap-12345678",
    "Volume2Type": "gp3",
    "Volume3Iops": "3000",
    "Volume3Throughput": "125",
    "Volume3KmsKeyId": "225bc21e-fc82-4388-8c91-855f3cadb63c",
    "Volume3Name": "/dev/xvdbd",
    "Volume3Size": "100",
    "Volume3Snapshot": "snap-12345678",
    "Volume3Type": "gp3",
    "Volume4Iops": "3000",
    "Volume4Throughput": "125",
    "Volume4KmsKeyId": "225bc21e-fc82-4388-8c91-855f3cadb63c",
    "Volume4Name": "/dev/xvdbe",
    "Volume4Size": "100",
    "Volume4Snapshot": "snap-12345678",
    "Volume4Type": "gp3",
    "Volume5Iops": "3000",
    "Volume5Throughput": "125",
    "Volume5KmsKeyId": "225bc21e-fc82-4388-8c91-855f3cadb63c",
    "Volume5Name": "/dev/xvdbf",
    "Volume5Size": "100",
    "Volume5Snapshot": "snap-12345678",
    "Volume5Type": "gp3"
  },
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-hrnfpt7l0qqumcelt"
}
```

EBS Volume | Create from Backup

Create an AWS Elastic Block Store (EBS) stack from backup.

Full classification: Deployment | Advanced stack components | EBS Volume | Create from backup

Change Type Details

Change type ID	ct-063qsm82cfxu6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required

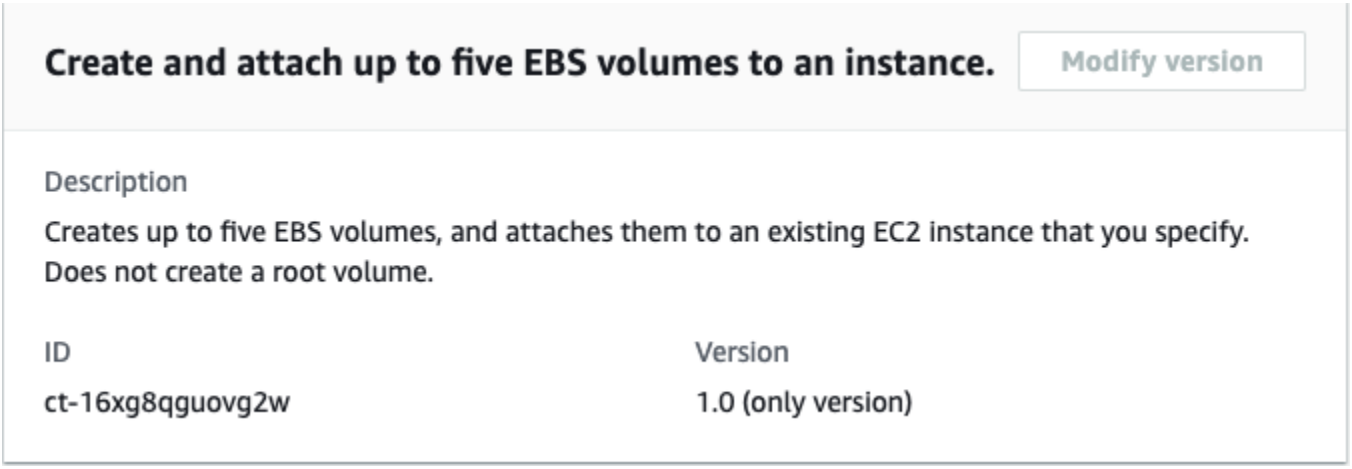
Customer approval	Not required
Execution mode	Automated

Additional Information

Create EBS volume from backup

Creating EBS with an AWS Backup with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating EBS with an AWS Backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-063qsm82cfxu6" \
--change-type-version "1.0" --title "EBS Create From Backup" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-StartRestoreJobEBS
\",\"Region\":\"us-east-1\",\"Parameters\":{\"AvailabilityZone\":[\"us-east-1a\"],
\"BackupVaultName\":[\"Default\"],\"RecoveryPointArn\":[\"arn:aws:ec2:us-
east-1::snapshot/snap-0000000000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it EbsCreateFromBackupParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-063qsm82cfxu6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EbsCreateFromBackupParams.json
   ```

2. Modify and save the EbsCreateFromBackupParams file.

   ```
   {
     "DocumentName": "AWSManagedServices-StartRestoreJobEBS",
     "Region": "us-east-1",
     "Parameters": {
       "AvailabilityZone": ["us-east-1a"],
       "BackupVaultName": ["Default"],
       "RecoveryPointArn": ["arn:aws:ec2:us-east-1::snapshot/snap-0000000000000000"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it EbsCreateFromBackupRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > EbsCreateFromBackupRfc.json
```

4. Modify and save the EbsCreateFromBackupRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-063qsm82cfxu6",
    "ChangeTypeVersion": "1.0",
    "Title": "EBS Create From Backup"
}
```

5. Create the RFC, specifying the EbsCreateFromBackupRfc file and the EbsCreateFromBackupParams file:

```
aws amscm create-rfc --cli-input-json file://EbsCreateFromBackupRfc.json --
execution-parameters file://EbsCreateFromBackupParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS, see Amazon Elastic Block Store (EBS).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-063qsm82cfxu6.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobEBS",
  "Region": "us-east-1",
  "Parameters": {
    "AvailabilityZone": ["us-east-1a"],
    "BackupVaultName": ["Vault01"],
    "RecoveryPointArn": ["arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobEBS",
  "Region": "us-east-1",
  "Parameters": {
    "AvailabilityZone": ["us-east-1a"],
    "BackupVaultName": ["Vault01"],
    "IOPS": ["250"],
    "RecoveryPointArn": ["arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"],
    "VolumeSize": ["100"],
    "VolumeType": ["gp3"],
    "Throughput": ["125"]
  }
}
```

EC2 Stack | Create

Use to create an Amazon Elastic Compute Cloud (EC2) instance.

Full classification: Deployment | Advanced stack components | EC2 stack | Create

Change Type Details

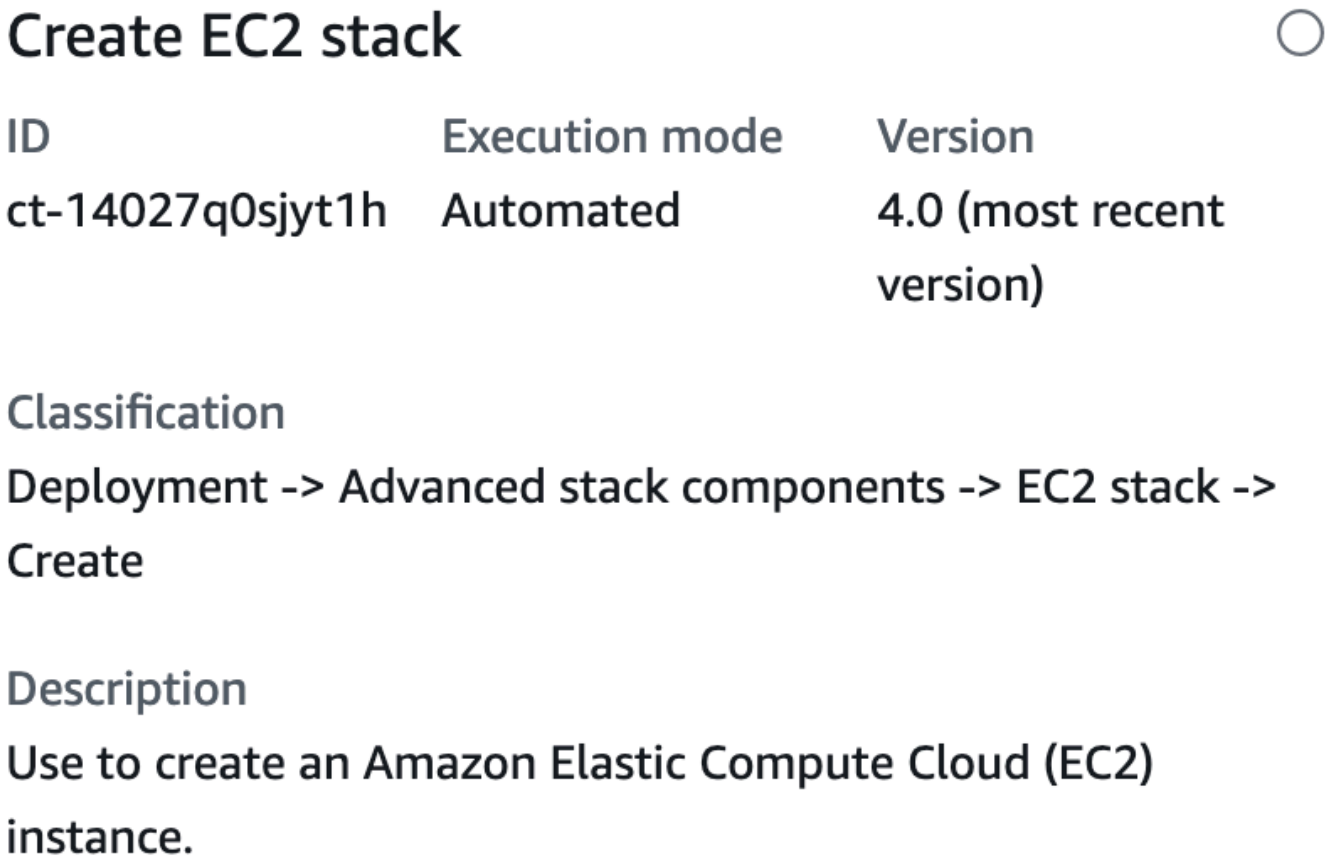
Change type ID	ct-14027q0sjyt1h
Current version	5.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create stack

Creating an EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-14027q0sjyt1h" --change-type-version "4.0"
 --title "EC2-Create-RFC" --execution-parameters "{\"Description\": \"Create a new
 EC2 Instance stack\",\"VpcId\": \"vpc-0a60eb65b4EXAMPLE\",\"Name\": \"My-EC2\",
\"TimeoutInMinutes\": 60,\"Parameters\": {\"InstanceAmiId\": \"ami-1234567890EXAMPLE\",
\"InstanceDetailedMonitoring\": false,\"InstanceEBSOptimized\": false,\"InstanceProfile
\": \"customer-mc-ec2-instance-profile\",\"InstanceRootVolumeIops\": 3000,
\"InstanceRootVolumeType\": \"gp3\",\"InstanceType\": \"t2.large\",\"InstanceUserData
\": \"\",\"InstanceSubnetId\": \"subnet-0bb1c79de3EXAMPLE\",\"EnforceIMDSV2\":
 \"false\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-14027q0sjyt1h" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateEC2Params.json
   ```

2. Modify and save the CreateEC2Params file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "Create a new EC2 Instance stack",
     "VpcId": "vpc-0a60eb65b4EXAMPLE",
   ```

```
  "Name": "My-EC2",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "InstanceAmiId": "ami-1234567890EXAMPLE",
    "InstanceDetailedMonitoring": false,
    "InstanceEBSOptimized": false,
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceRootVolumeIops": 3000,
    "InstanceRootVolumeType": "gp3",
    "InstanceType": "t2.large",
    "InstanceUserData": "",
    "InstanceSubnetId": "subnet-0bb1c79de3EXAMPLE",
    "EnforceIMDSV2": "false"
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateEC2Rfc.json
```

4. Modify and save the CreateEC2Rfc.json file. For example, you can replace the contents with something like this:.

```
{
"ChangeTypeVersion":    "4.0",
"ChangeTypeId":         "ct-14027q0sjyt1h",
"Title":                "EC2-Create-RFC"
}
```

5. Create the RFC, specifying the CreateEC2Rfc file and the CreateEC2Params file:

```
aws amscm create-rfc --cli-input-json file://CreateEC2Rfc.json  --execution-
parameters file://CreateEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Security Groups

Starting with version 3.0 of this change type, AMS does not attach the default AMS security groups if you specify your own security groups. If you do not specify your own security groups in the request, AMS attaches the AMS default security groups. In previous versions, AMS attached the default security groups whether or not you provided your own security groups.

Currently, if you specify custom security groups, you must also specify the IDs of the default AMS security groups for your account, `mc-initial-garden-SG-name` and `mc-initial-garden-SG-name`.

Instance Types

AMS does not recommend the **t2.micro/t3.micro** and **t2.nano/t3.nano** types. These are smaller instance types, and can degrade the performance of your application and AMS tools. EC2 instances need enough capacity to support AMS tools such as EPS, SSM, and Cloudwatch in addition to the application workload. For more information, see Choosing the Right EC2 Instance Type for Your Application.

To create an EC2 stack with additional volumes, see EC2 Stack | Create (with Additional Volumes).

You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

If needed, see EC2 instance stack create fail.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-14027q0sjyt1h.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
```

```
  "Name": "Test Stack",
  "TimeoutInMinutes": 360,
  "Parameters": {
    "InstanceAmiId": "ami-1234567890abcdef0",
    "InstanceSubnetId": "subnet-1234567890abcdef0",
    "EnforceIMDSV2": "true"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "InstanceAmiId": "ami-a0b1c2d3",
    "InstanceDetailedMonitoring": false,
    "InstanceEBSOptimized": false,
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceRootVolumeIops": 3000,
    "InstanceRootVolumeName": "/dev/xvda",
    "InstanceRootVolumeSize": 60,
    "InstanceRootVolumeType": "gp3",
    "InstancePrivateStaticIp": "172.16.0.0",
    "InstanceSubnetId": "subnet-a0b1c2d3",
    "InstanceType": "t2.large",
    "InstanceUserData": "pwd\nls -ltrh\necho \"Hello, World\"",
    "EnforceIMDSV2": "true"
  }
}
```

EC2 Stack | Create (With Additional Volumes)

Create an Amazon Elastic Compute Cloud (EC2) instance with up to five additional volumes.

Full classification: Deployment | Advanced stack components | EC2 stack | Create (with additional volumes)

Change Type Details

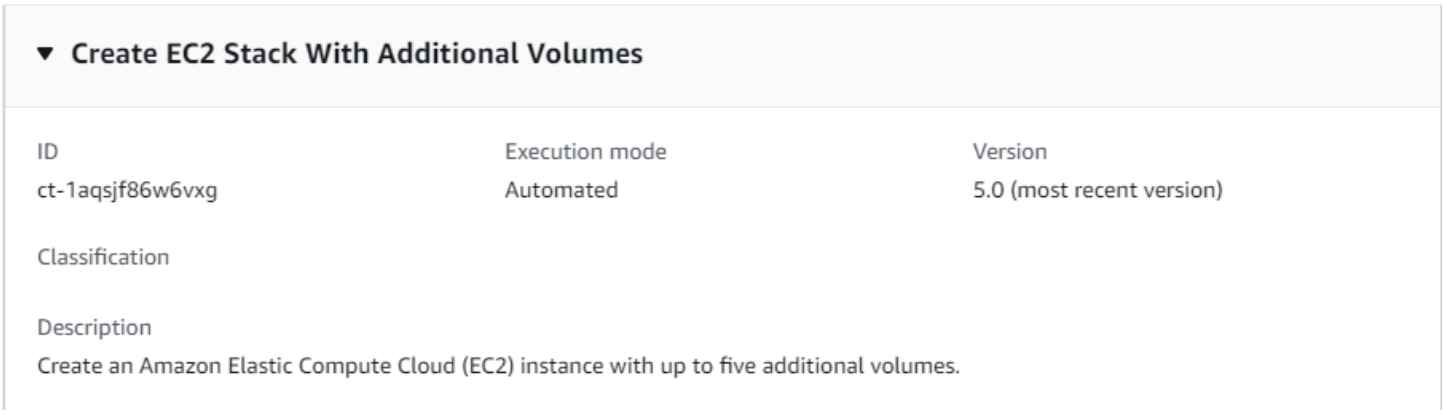
Change type ID	ct-1aqsjf86w6vxg
Current version	5.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create stack (with additional volumes)

Creating an EC2 instance and additional volumes with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an EC2 instance and additional volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID (example shows required parameters only). For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1aqsjf86w6vxg" --change-type-version "4.0"
 --title "EC2-Create-A-V-QC" --execution-parameters "{\"Description\":\"My EC2 stack
 with addl vol\",\"VpcId\":\"VPC_ID\",\"Name\":\"My Stack\",\"StackTemplateId\":
\"stm-nn8v8ffhcal611bmo\",\"TimeoutInMinutes\":60,\"Parameters\":{\"InstanceAmiId\":
\"AMI_ID\",\"InstanceSubnetId\":\"SUBNET_ID\"}}
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateEC2AVParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1aqsjf86w6vxg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateEC2AVParams.json
   ```

2. Modify and save the CreateEC2AVParams file (example shows most parameters). For example, you can replace the contents with something like this:

```
{
"Description":      "EC2-Create-1-Addl-Volumes",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-nn8v8ffhcal611bmo",
"Name":             "My-EC2-1-Addl-Volume",
"TimeoutInMinutes": 60,
"Parameters":   {
    "InstanceAmiId":    "AMI_ID",
    "InstanceSecurityGroupIds": "SECURITY_GROUP_ID",
    "InstanceCoreCount": 1,
    "InstanceThreadsPerCore": 2,
    "InstanceDetailedMonitoring": "true",
    "InstanceEBSOptimized": "false",
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceRootVolumeIops": 100,
    "InstanceRootVolumeName": "/dev/xvda",
    "InstanceRootVolumeSize": 50,
    "InstanceRootVolumeType": "io1",
    "RootVolumeKmsKeyId": "default",
    "InstancePrivateStaticIp": "10.27.0.100",
    "InstanceSecondaryPrivateIpAddressCount": 0,
    "InstanceTerminationProtection": "false",
    "InstanceType": "t3.large",
    "CreditSpecification": "unlimited",
    "InstanceUserData": "echo $",
    "Volume1Encrypted": "true",
    "Volume1Iops":      "IOPS"
    "Volume1KmsKeyId":  "KMS_MASTER_KEY_ID",
    "Volume1Name":      "xvdh"
    "Volume1Size":      "2 GiB",
    "Volume1Snapshot":  "SNAPSHOT_ID",
    "Volume1Type":      "iol",
    "InstanceSubnetId": "SUBNET_ID"
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateEC2AVRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateEC2AVRfc.json
```

4. Modify and save the CreateEC2AVRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "4.0",
"ChangeTypeId":         "ct-1aqsjf86w6vxg",
"Title":                "EC2-Create-1-Addl-Volume-RFC"
}
```

5. Create the RFC, specifying the CreateEC2AVRfc file and the CreateEC2AVParams file:

```
aws amscm create-rfc --cli-input-json file://CreateEC2AVRfc.json  --execution-
parameters file://CreateEC2AVParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Important

There is a new version of this change type, v 4.0, that uses a different StackTemplateId (stm-nn8v8ffhcal611bmo). This is important if you're submitting the RFC with this change type at the command line. The new version introduces two new parameters (**RootVolumeKmsKeyId** and **CreditSpecification**) and changes the default for one existing parameter (**InstanceType**).

Instance Types

- If you choose to specify the number of cores or threads, you must specify values for both. Use the parameters `InstanceCoreCount` and `InstanceThreadsPerCore`. To find valid combinations of cores/threads, see CPU cores and threads per CPU core per instance type .

- AMS does not recommend the **t2.micro/t3.micro** or **t2.nano/t3.nano** instance types. These are too small to support AMS tools such as EPS, SSM, and Cloudwatch in addition to your business workload. For more information, see Choosing the Right EC2 Instance Type for Your Application.
- In version 4.0, the default type was raised from **t2.large** to **t3.large**. T3 instances launch with 'unlimited credits' by default. You won't experience CPU throttling even if the instance consumes all CPU credits. You can, instead, choose T2 instances and use the CreditSpecification unlimited option.
- For more information about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

To update your EC2 stack with additional volumes after they're created, see EC2 Instance stack: Updating (With Additional Volumes)

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1aqsjf86w6vxg.

Example: Required Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-nn8v8ffhcal611bmp",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "InstanceAmiId" : "ami-1234567890abcdef0",
    "InstanceSubnetId" : "subnet-1234567890abcdef0",
    "EnforceIMDSV2": "true"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
```

```
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "Parameters": {
    "InstanceAmiId": "ami-12345678",
    "InstanceCoreCount": 0,
    "InstanceThreadsPerCore": 0,
    "InstanceRootVolumeName": "/dev/xvda",
    "InstanceRootVolumeSize": 100,
    "InstanceSubnetId": "subnet-12345678",
    "InstanceDetailedMonitoring": "false",
    "InstanceEBSOptimized": "false",
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceRootVolumeIops": 1000,
    "InstanceRootVolumeType": "io1",
    "InstancePrivateStaticIp": "172.16.0.10",
    "InstanceSecondaryPrivateIpAddressCount" : 1,
    "InstanceTerminationProtection" : "true",
    "InstanceType": "t2.small",
    "InstanceUserData": "#!/bin/bash\\npwd\\nls -ltrh\\necho \"Hello, World\"",
    "Volume1Iops": 100,
    "Volume1KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Volume1Name": "/dev/sdf",
    "Volume1Size": 100,
    "Volume1Type": "io1",
    "Volume2Iops": 100,
    "Volume2KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Volume2Name": "/dev/sdg",
    "Volume2Size": 100,
    "Volume2Type": "io1",
    "Volume3Iops": 100,
    "Volume3KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Volume3Name": "/dev/sdh",
    "Volume3Size": 100,
    "Volume3Type": "io1",
    "Volume4Iops": 100,
```

```
    "Volume4KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Volume4Name": "/dev/sdi",
    "Volume4Size": 100,
    "Volume4Type": "io1",
    "Volume5Iops": 100,
    "Volume5KmsKeyId": "12345678-1234-1234-1234-1234567890ab",
    "Volume5Name": "/dev/sdj",
    "Volume5Size": 100,
    "Volume5Type": "io1",
    "EnforceIMDSV2": "true"
  },
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-nn8v8ffhcal611bmp"
}
```

Elastic File System (EFS) | Create

Use to create a Elastic File System (EFS) stack

Full classification: Deployment | Advanced stack components | Elastic File System (EFS) | Create

Change Type Details

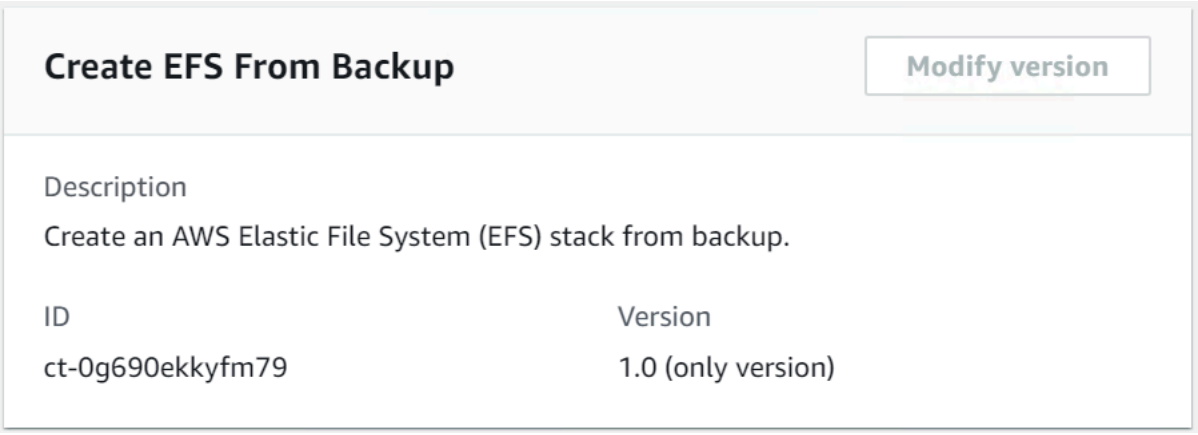
Change type ID	ct-2uw99b8hpncnu
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create EFS

Creating an Elastic File System with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Elastic File System with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 ct-2uw99b8hpncnu --change-type-version 1.0 --title "TEST-EFS-RFC" --description
 "TEST-EFS-RFC" --execution-parameters "{\"Description\":\"TEST-EFS\",\"Name\":\"Test-
EFS-Stack\",\"VpcId\":\"VPC_ID\",\"TimeoutInMinutes\":60,\"Parameters\":{\"Encrypted
\":true,\"PerformanceMode\":\"generalPurpose\",\"MountTargets\":[{\"AvailabilityZone\":
\"us-east-1a\"}]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateEfsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2uw99b8hpncnu" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateEfsParams.json
   ```

2. Modify and save the CreateEfsParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "EFS-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-sdhopv00000000000",
   "Name":             "My-EFS",

   "Parameters":   {
       "ELBSubnetIds":  ["PUBLIC_SUBNET"],
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it CreateEfsRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateEfsRfc.json
   ```

4. Modify and save the CreateEfsRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-2uw99b8hpncnu",
   "Title":                "EFS-Create-RFC"
   ```

```
}
```

5. Create the RFC, specifying the CreateEfsRfc file and the CreateEfsParams file:

```
aws amscm create-rfc --cli-input-json file://CreateEfsRfc.json --execution-
parameters file://CreateEfsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> Currently AMS has a known issue that is preventing the auto-mounting of the EFS by using the UserData field when creating new instances.

For more information, see Amazon Elastic File System Documentation.

This change type creates an EFS and mounts targets in your specified Availability Zones. You only need to specify an Availability Zone to create a mount target, but you can optionally specify a specific subnet to create the mount target in, and a private IP address to give the mount target within that subnet. If you only specify an Availability Zone, AMS picks a subnet/IP address to give the mount target. For an example of creating an EFS, , see Create EFS.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2uw99b8hpncnu.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "VpcId": "vpc-1234567890abcdef0",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "Encrypted": true,
```

```
    "PerformanceMode": "generalPurpose",
    "MountTargets": [
      {
        "AvailabilityZone": "us-east-1a"
      }
    ]
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "VpcId": "vpc-12345678",
  "TimeoutInMinutes": 60,
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "Parameters": {
    "Encrypted": true,
    "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "PerformanceMode": "generalPurpose",
    "MountTargets": [
      {
        "AvailabilityZone": "us-east-1a",
        "SubnetId": "subnet-12345678",
        "IpAddress": "10.0.0.1"
      }
    ]
  }
}
```

Elastic File System (EFS) | Create from Backup

Create an AWS Elastic File System (EFS) stack from backup.

Full classification: Deployment | Advanced stack components | Elastic File System (EFS) | Create from backup

Change Type Details

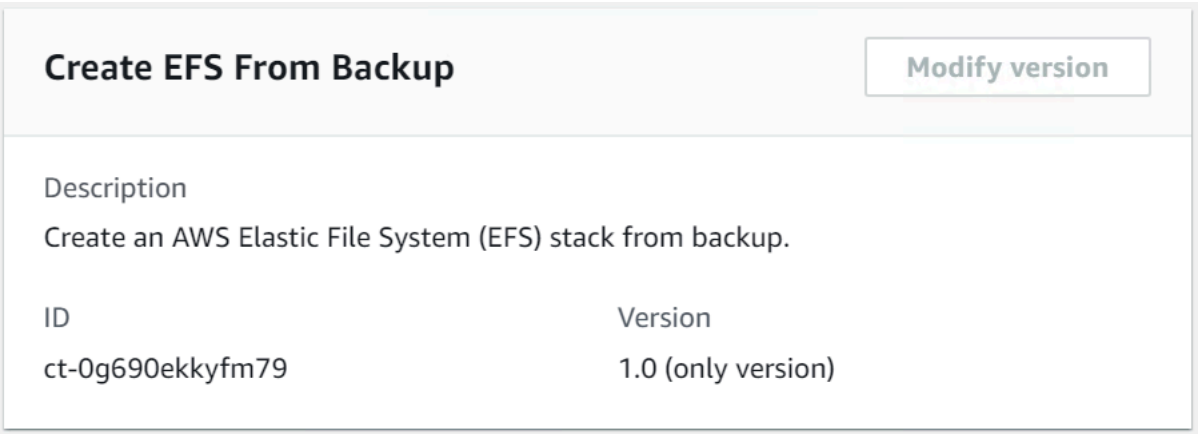
Change type ID	ct-0g690ekkyfm79
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create EFS from backup

Creating EFS with an AWS Backup with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating EFS with an AWS Backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-0g690ekkyfm79" \
--change-type-version "1.0" --title "EFS Create From Backup" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-StartRestoreJobEFS
\",\"Region\":\"us-east-1\",\"Parameters\":{\"BackupVaultName\":[\"Default\"],
\"RecoveryPointArn\":[\"RECOVERY_POINT_ARN\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it EfsCreateFromBackupParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0g690ekkyfm79"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EfsCreateFromBackupParams.json
   ```

2. Modify and save the EfsCreateFromBackupParams file.

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-StartRestoreJobEFS",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["Default"],
    "EnableEncryption": ["true"],
    "KmsKeyId": ["arn:aws:kms:us-
east-1:000000000000:key/00000000-0000-0000-0000-000000000000"],
    "PerformanceMode": ["maxIO"],
    "RecoveryPointArn": ["arn:aws:backup:us-east-1:000000000000:recovery-
point:00000000-0000-0000-0000-000000000000"],
    "RestoreToNewFileSystem": ["true"]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it EfsCreateFromBackupRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > EfsCreateFromBackupRfc.json
```

4. Modify and save the EfsCreateFromBackupRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-0g690ekkyfm79",
    "ChangeTypeVersion": "1.0",
    "Title": "EFS Create From Backup"
}
```

5. Create the RFC, specifying the EfsCreateFromBackupRfc file and the EfsCreateFromBackupParams file:

```
aws amscm create-rfc --cli-input-json file://EfsCreateFromBackupRfc.json --
execution-parameters file://EfsCreateFromBackupParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0g690ekkyfm79.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobEFS",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["Vault01"],
    "RecoveryPointArn": ["arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobEFS",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["Vault01"],
    "EnableEncryption": ["true"],
    "ItemsToRestore": ["/dir1", "/dir2"],
    "KmsKeyId": ["arn:aws:kms:us-
east-1:000000000000:key/00000000-0000-0000-0000-000000000000"],
    "PerformanceMode": ["maxIO"],
    "RecoveryPointArn": ["arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"],
    "RestoreToNewFileSystem": ["true"]
  }
}
```

Identity and Access Management (IAM) | Create Access Key

Create a new AWS secret access key and corresponding AWS access key ID for the specified user.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create access key

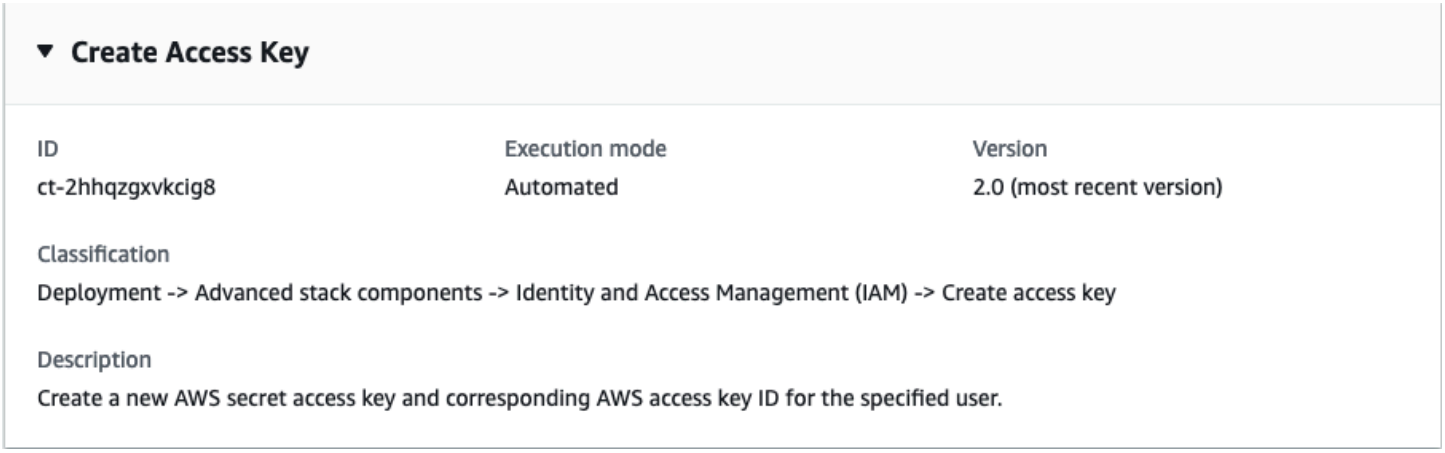
Change Type Details

Change type ID	ct-2hhqzgxvkcig8
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create access key

Creating access key with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating access key with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Note

When pasting in a policy document, note that the RFC only accepts policy pastes up to 5,000 characters. If your file has more than 5,000 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2hhqzgxvkcig8" --change-type-version
 "2.0" --title "Create access key" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-CreateIAMAccessKey\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"UserARN\": \"arn:aws:iam::012345678910:user/myusername\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamAccessKeyParameters.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hhqzgxvkcig8"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamAccessKeyParameters.json
   ```

2. Modify and save the CreateIamAccessKeyParameters.json file; example creates an IAM Role with policy documents pasted inline.

```
{
    "DocumentName": "AWSManagedServices-CreateIAMAccessKey",
    "Region": "ap-southeast-2",
    "Parameters": {
        "UserARN": "arn:aws:iam::012345678910:user/myusername"
    }
  }
}
```

3. Output the RFC template JSON file to a file named CreateIamAccessKeyRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamAccessKeyRfc.json
```

4. Modify and save the CreateIamAccessKeyRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-2hhqzgxvkcig8",
  "Title": "Create IAM access key"
}
```

5. Create the RFC, specifying the CreateIamAccessKeyRfc.json file and the CreateIamAccessKeyParameters.json file:

```
aws amscm create-rfc --cli-input-json file://CreateIamAccessKeyRFC.json --
execution-parameters file://CreateIamAccessKeyParameters.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2hhqzgxvkcig8.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateIAMAccessKeyV2",
  "Region": "us-east-1",
  "Parameters": {
    "UserARN": "arn:aws:iam::012345678910:user/myusername"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateIAMAccessKeyV2",
  "Region": "us-east-1",
  "Parameters": {
    "UserARN": "arn:aws:iam::012345678910:user/myusername"
  }
}
```

Identity and Access Management (IAM) | Create Account Alias

Create an AWS account alias. Note that an AWS account can have only one alias. This operation fails if the AWS account already has an alias. To update an existing account alias, use the Update Account Alias (ct-3skaisgnq0pf8) change type.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create account alias

Change Type Details

Change type ID	ct-36x3u7v2oklwd
Current version	1.0
Expected execution duration	60 minutes

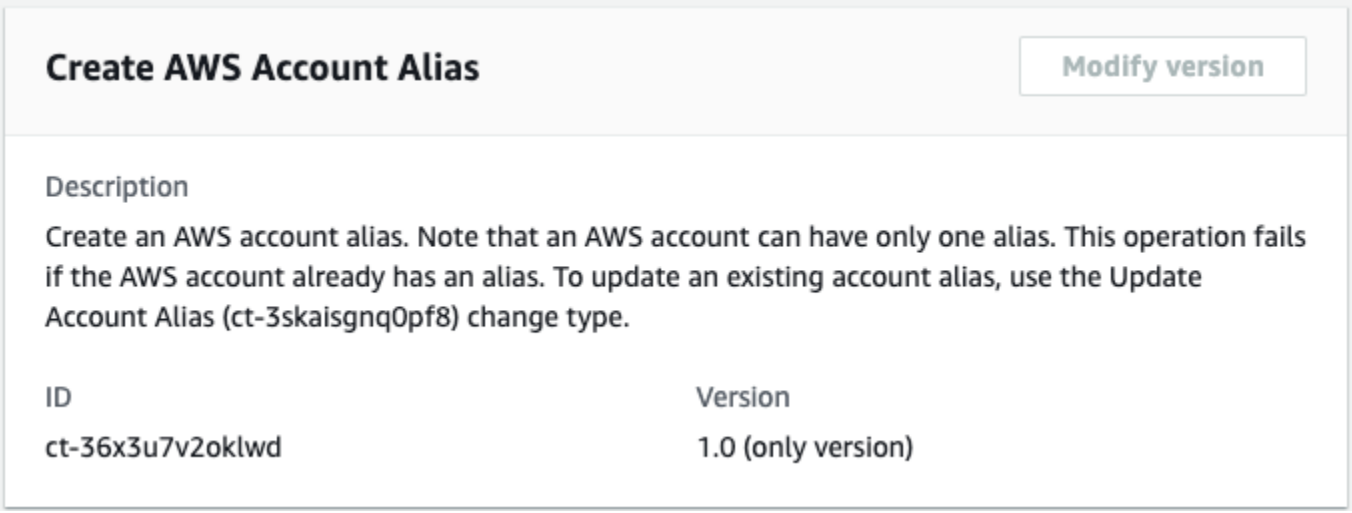
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM account alias

Creating IAM account alias with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM account alias with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-36x3u7v2oklwd" --change-type-
version "1.0" --title "Create Account Alias" --execution-parameters
 '{"DocumentName":"AWSManagedServices-CreateAccountAlias","Region":"us-
east-1","Parameters":{"AWSAccountAlias":["my-alias"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamAccountAliasParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-36x3u7v2oklwd"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamAccountAliasParams.json
   ```

2. Modify and save the CreateIamAccountAliasParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
       "DocumentName": "AWSManagedServices-CreateAccountAlias",
       "Region": "us-east-1",
       "Parameters": {
           "AWSAccountAlias": [
           "my-alias"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file named CreateIamAccountAliasRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamAccountAliasRfc.json
```

4. Modify and save the CreateIamAccountAliasRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-36x3u7v2oklwd",
    "ChangeTypeVersion": "1.0",
    "Title": "Create Account Alias"
}
```

5. Create the RFC, specifying the CreateIamAccountAliasRfc file and the CreateIamAccountAliasParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamAccountAliasRfc.json  --
execution-parameters file://CreateIamAccountAliasParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-36x3u7v2oklwd.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateAccountAlias",
  "Region": "us-east-1",
```

```
  "Parameters": {
    "AWSAccountAlias": ["myalias"]
  }
}
```

Identity and Access Management (IAM) | Create EC2 Instance Profile

Create an IAM instance profile to use with EC2 instances. Each ARN specified in the parameters creates a part of the IAM policy. Use the Preview option to see what the completed, generated, policy looks like before it is created and implemented.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create EC2 instance profile

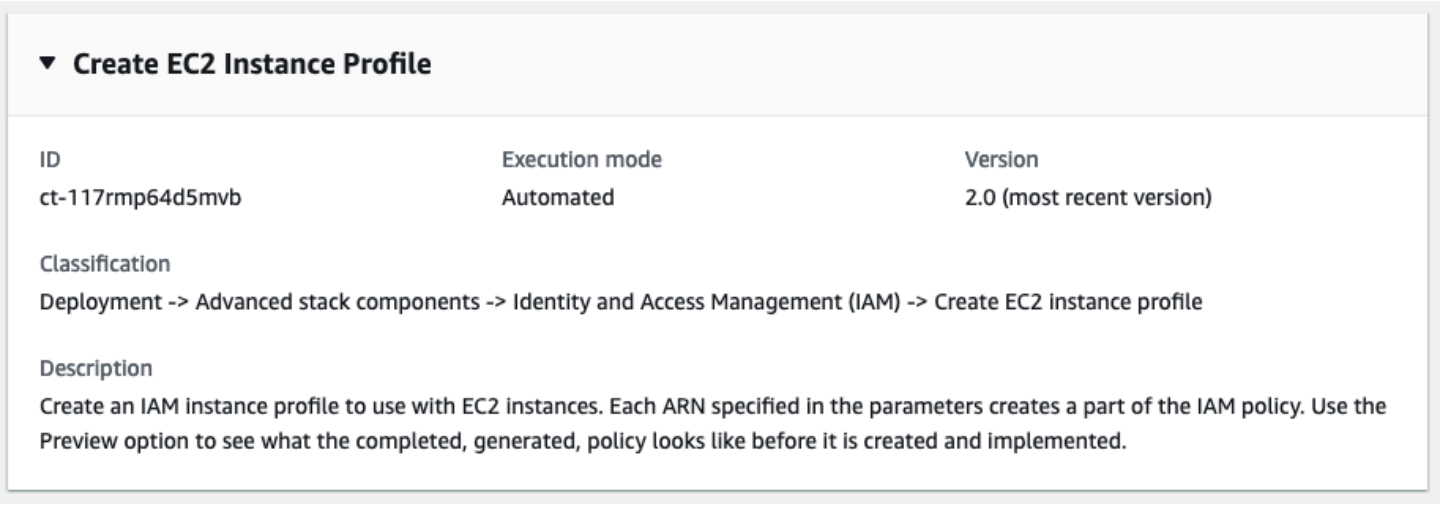
Change Type Details

Change type ID	ct-117rmp64d5mvb
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM EC2 profile

Creating IAM EC2 profiles with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM EC2 profiles with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE (required parameters only):

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-117rmp64d5mvb" --change-type-version
 "2.0" --title "new EC2 instance profile" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-HandleCreateIAMRole-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": { \"RoleName\": \"customer_application_instance_profile\",
 \"ServicePrincipal\": \"ec2.amazonaws.com\", \"Preview\": \"No\" }}"
```

TEMPLATE CREATE (all parameters):

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamEc2ProfileParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-117rmp64d5mvb"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamEc2ProfileParams.json
   ```

2. Modify and save the CreateIamEc2ProfileParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName": "AWSManagedServices-HandleCreateIAMRole-Admin",
     "Region": "us-east-1",
     "Parameters": {
     "RoleName": "customer_application_instance_profile",
     "ServicePrincipal": "ec2.amazonaws.com",
     "Preview": "No"
     }
   }
   ```

3. Output the RFC template JSON file to a file named CreateIamEc2ProfileRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateIamEc2ProfileRfc.json
   ```

4. Modify and save the CreateIamEc2ProfileRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion": "2.0",
"ChangeTypeId": "ct-117rmp64d5mvb",
"Title": "Create New EC2 Instance Profile"
}
```

5. Create the RFC, specifying the CreateIamEc2ProfileRfc file and the CreateIamEc2ProfileParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamEc2ProfileRfc.json  --
execution-parameters file://CreateIamEc2ProfileParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-117rmp64d5mvb.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateIAMRole-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "RoleName": "customer_application_instance_profile",
    "ServicePrincipal": "ec2.amazonaws.com",
    "Preview": "No"
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateIAMRole-Admin",
  "Region" : "us-east-1",
  "Parameters": {
```

```
    "CloudWatchAlarmReadAccess": ["arn:aws:cloudwatch:us-
east-1:123456789012:alarm:myalarm*"],
    "CloudWatchAlarmWriteAccess": ["arn:aws:cloudwatch:us-
east-1:123456789012:alarm:myalarm*"],
    "CloudWatchLogsReadAccess": ["arn:aws:logs:us-east-1:123456789012:log-
group:myparam*:log-stream:mylogstream"],
    "CloudWatchLogsWriteAccess": ["arn:aws:logs:us-east-1:123456789012:log-
group:mylogs*"],
    "CloudWatchMetricsReadAccess": ["*"],
    "CloudWatchMetricsWriteAccess": ["Company/AppMetric"],
    "DynamoDBDataReadWriteAccess": ["arn:aws:dynamodb:us-east-1:123456789012:table/
mytable*"],
    "DynamoDBResourceReadAccess": ["arn:aws:dynamodb:us-east-1:123456789012:table/
anotherTable"],
    "KMSCryptographicOperationAccess": ["arn:aws:kms:us-
east-1:123456789012:key/97f43232-6bdc-4830-b54c-2d2926ba69aa"],
    "KMSReadAccess": ["arn:aws:kms:us-east-1:123456789012:key/97f43232-6bdc-4830-
b54c-2d2926ba69aa"],
    "Preview": "No",
    "RoleName": "customer_application_instance_profile",
    "RolePath": "/test/",
    "S3ReadAccess": ["arn:aws:s3:::my-s3-us-east-1/*"],
    "S3WriteAccess": ["arn:aws:s3:::my-s3-ap-southeast-2/developers/design_info.doc"],
    "SNSReadAccess": ["arn:aws:sns:us-east-1:123456789012:mytopic*"],
    "SNSWriteAccess": ["arn:aws:sns:us-east-1:123456789012:MyTopic*"],
    "SQSReadAccess": ["arn:aws:sqs:us-east-1:123456789012:Myqueue*"],
    "SQSWriteAccess": ["arn:aws:sqs:us-east-1:123456789012:MyQueeu*"],
    "SSMReadAccess": ["arn:aws:ssm:us-east-1:123456789012:parameter/myparam*"],
    "SSMWriteAccess": ["arn:aws:ssm:us-east-1:123456789012:parameter/myparam*"],
    "STSAssumeRole": ["arn:aws:iam::123456789012:role/roleName"],
    "SecretsManagerReadAccess": ["arn:aws:secretsmanager:us-
east-1:123456789012:secret:mysecret*"],
    "ServicePrincipal": "ec2.amazonaws.com",
    "AdditionalPolicy" : "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Action\":[\"iam:ListRoles\",\"iam:ListAccountAliases\"],\"Resource\":\"*
\"}]}"
  }
}
```

Identity and Access Management (IAM) | Create Entity or Policy (Read-Write Permissions)

Create Identity and Access Management (IAM) role or policy with read-write permissions. You must have enabled this feature with change type ct-1706xvvk6j9hf before submitting this request. Automated IAM provisioning with read-write permissions runs over 200 validations to help ensure successful outcomes.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create entity or policy (read-write permissions)

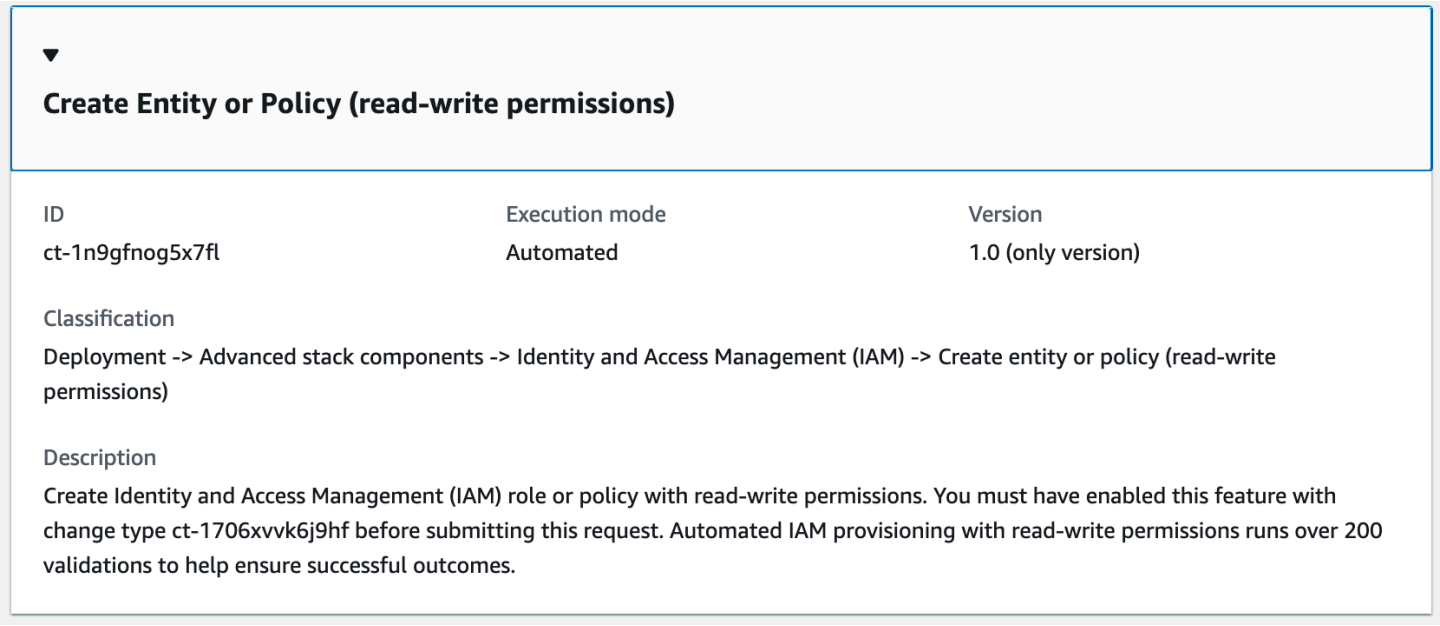
Change Type Details

Change type ID	ct-1n9gfnog5x7fl
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Automated

Additional Information

Create IAM entity or policy

Creating IAM entity or policy with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM entity or policy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1n9gfnog5x7fl" --change-type-
version "1.0" --title "Create role or policy" --execution-parameters
 '{"DocumentName":"AWSManagedServices-HandleAutomatedIAMProvisioningCreate-
Admin","Region":"us-east-1","Parameters":{"ValidateOnly":"No"},"RoleDetails":
{"Roles":[{"RoleName":"RoleTest01","Description":"This is a test
 role","AssumeRolePolicyDocument":"{"Version":"2012-10-17","Statement":[{"Effect":"Allow","Prin
{"AWS":"arn:aws:iam::123456789012:root"},"Action":"sts:AssumeRole"}]}","ManagedPolicyArns":
["arn:aws:iam::123456789012:policy/policy01","arn:aws:iam::123456789012:policy/
policy02"],"Path":"/","MaxSessionDuration":"7200","PermissionsBoundary":"arn:aws:iam::12345678
permission_boundary01","InstanceProfile":"No"}]},"ManagedPolicyDetails":
{"Policies":[{"ManagedPolicyName":"TestPolicy01","Description":"This is customer
 policy","Path":"/test/","PolicyDocument":"{"Version":"2012-10-17","Statement":
[{"Sid":"AllQueueActions","Effect":"Allow","Action":"sqs:ListQueues","Resource":"*","Condition"
{"ForAllValues:StringEquals":{"aws:tagKeys":["temporary"]}}}]}"}]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1n9gfnog5x7fl"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamResourceParams.json
   ```

2. Modify and save the CreateIamResourceParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName": "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin",
     "Region": "us-east-1",
     "Parameters": {
   ```

```
    "ValidateOnly": "No"
  },
  "RoleDetails": {
    "Roles": [
      {
        "RoleName": "RoleTest01",
        "Description": "This is a test role",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}",
        "ManagedPolicyArns": [
          "arn:aws:iam::123456789012:policy/policy01",
          "arn:aws:iam::123456789012:policy/policy02"
        ],
        "Path": "/",
        "MaxSessionDuration": "7200",
        "PermissionsBoundary": "arn:aws:iam::123456789012:policy/
permission_boundary01",
        "InstanceProfile": "No"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "Description": "This is customer policy",
        "Path": "/test/",
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource
\":\"*\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary
\"]}}}]}"
      }
    ]
  }
}
```

3. Output the RFC template JSON file to a file named CreateIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamResourceRfc.json
```

4. Modify and save the CreateIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1n9gfnog5x7fl",
  "Title": "Create entity or policy (read-write permissions)"
}
```

5. Create the RFC, specifying the CreateIamResourceRfc file and the CreateIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamResourceRfc.json  --
execution-parameters file://CreateIamResourceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- After an IAM role is provisioned in your account, depending on the role and the policy document you attach to the role, you may need to onboard the role in your federation solution.
- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1n9gfnog5x7fl.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "ValidateOnly": "No"
  },
  "RoleDetails": {
    "Roles": [
      {
```

```
        "RoleName": "RoleTest01",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "Description": "This is customer policy",
        "Path": "/test/",
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource\":\"*
\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary\"]}}}]}"
      }
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "ValidateOnly": "No"
  },
  "RoleDetails": {
    "Roles": [
      {
        "RoleName": "RoleTest01",
        "Description": "This is a test role",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}",
        "ManagedPolicyArns": [
          "arn:aws:iam::123456789012:policy/policy01",
          "arn:aws:iam::123456789012:policy/policy02"
        ],
        "Path": "/",
        "MaxSessionDuration": "7200",
```

```
        "PermissionsBoundary": "arn:aws:iam::123456789012:policy/
permission_boundary01",
        "InstanceProfile": "No"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "Description": "This is customer policy",
        "Path": "/test/",
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource\":\"*
\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary\"]}}}]}"
      }
    ]
  }
}
```

Identity and Access Management (IAM) | Create Entity or Policy (Review Required)

Create Identity and Access Management (IAM) user, role, or policy.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create entity or policy (review required)

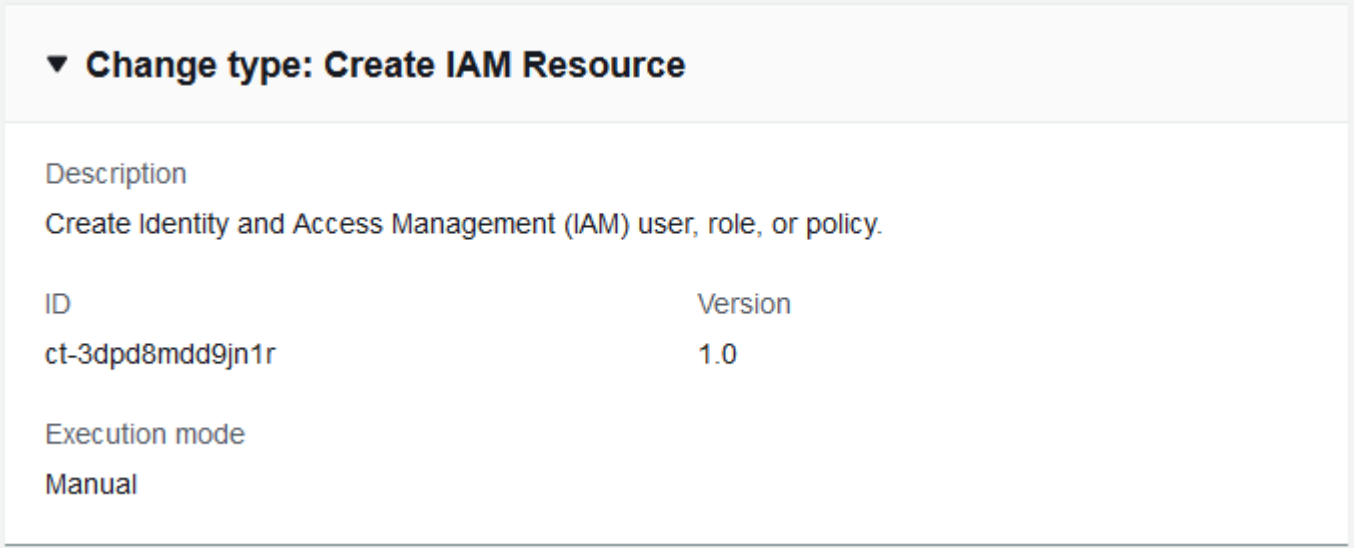
Change Type Details

Change type ID	ct-3dpd8mdd9jn1r
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create IAM entity or policy (review required)

Creating IAM resources (review required) with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM resources (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

> **Note**
>
> When pasting in a policy document, note that the RFC only accepts policy pastes up to 20,480 characters. If your file has more than 20,480 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3dpd8mdd9jn1r" --change-type-version "1.0"
 --title "TestIamCreate" --execution-parameters "{\"UseCase\":\"IAM_RESOURCE_DETAILS\",
\"IAM Role\":[{\"RoleName\":\"ROLE_NAME\",\"TrustPolicy\":\"TRUST_POLICY\",
\"RolePermissions\":\"ROLE_PERMISSIONS\"}],\"Operation\":\"Create\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3dpd8mdd9jn1r"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamResourceParams.json
   ```

2. Modify and save the CreateIamResourceParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "UseCase": "IAM_RESOURCE_DETAILS",
     "IAM Role": [
       {
         "RoleName": "codebuild_ec2_test_role",
         "TrustPolicy": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Effect\":
   \"Allow\",\"Principal\":{\"Service\":\"codebuild.amazonaws.com\"},\"Action\":
   \"sts:AssumeRole\"}]}",
         "RolePermissions": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
   \"Allow\",\"Action\":[\"ec2:DescribeInstanceStatus\"],\"Resource\":\"*\"}]}"
   ```

```
    }
  ],
  "Operation": "Create"
}
```

3. Output the RFC template JSON file to a file named CreateIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamResourceRfc.json
```

4. Modify and save the CreateIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-3dpd8mdd9jn1r",
"Title": "Create IAM Role"
}
```

5. Create the RFC, specifying the CreateIamResourceRfc file and the CreateIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamResourceRfc.json  --
execution-parameters file://CreateIamResourceParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- After an IAM role is provisioned in your account, you must onboard the role in your federation solution.
- When pasting in a policy document, note that the RFC only accepts policy pastes up to 20,480 characters. If your policy has more than 20,480 characters, create a service request to upload the policy, and then refer to that service request in the RFC that you open for IAM.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3dpd8mdd9jn1r.

Example: Required Parameters

```
{
  "UseCase": "Use case...",
  "Operation": "Create"
}
```

Example: All Parameters

```
{
  "UseCase": "Use case...",
  "IAM User": [
    {
      "UserName": "user-a",
      "AccessType": "Console access",
      "UserPermissions": "Power User permissions",
      "Tags": [
        {
          "Key": "foo",
          "Value": "bar"
        },
        {
          "Key": "testkey",
          "Value": "testvalue"
        }
      ]
    }
  ],
  "IAM Role": [
    {
      "RoleName": "role-b",
      "TrustPolicy": "Trust policy example",
      "RolePermissions": "Role permissions example",
```

```
      "Tags": [
        {
          "Key": "foo",
          "Value": "bar"
        },
        {
          "Key": "testkey",
          "Value": "testvalue"
        }
      ]
    }
  ],
  "IAM Policy": [
    {
      "PolicyName": "policy1",
      "PolicyDocument": "Policy document example 1",
      "RelatedResources": [
        "resourceA",
        "resourceB"
      ]
    },
    {
      "PolicyName": "policy2",
      "PolicyDocument": "Policy document example 2",
      "RelatedResources": [
        "resourceC",
        "resourceD"
      ]
    }
  ],
  "Operation": "Create",
  "Priority": "Medium"
}
```

Identity and Access Management (IAM) | Create Lambda Execution Role

Create an Lambda execution role to use with Lambda Function. Each ARN specified in the parameters creates a part of the IAM policy. Use the Preview option to see what the completed, generated, policy looks like before it is created and implemented.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Lambda execution role

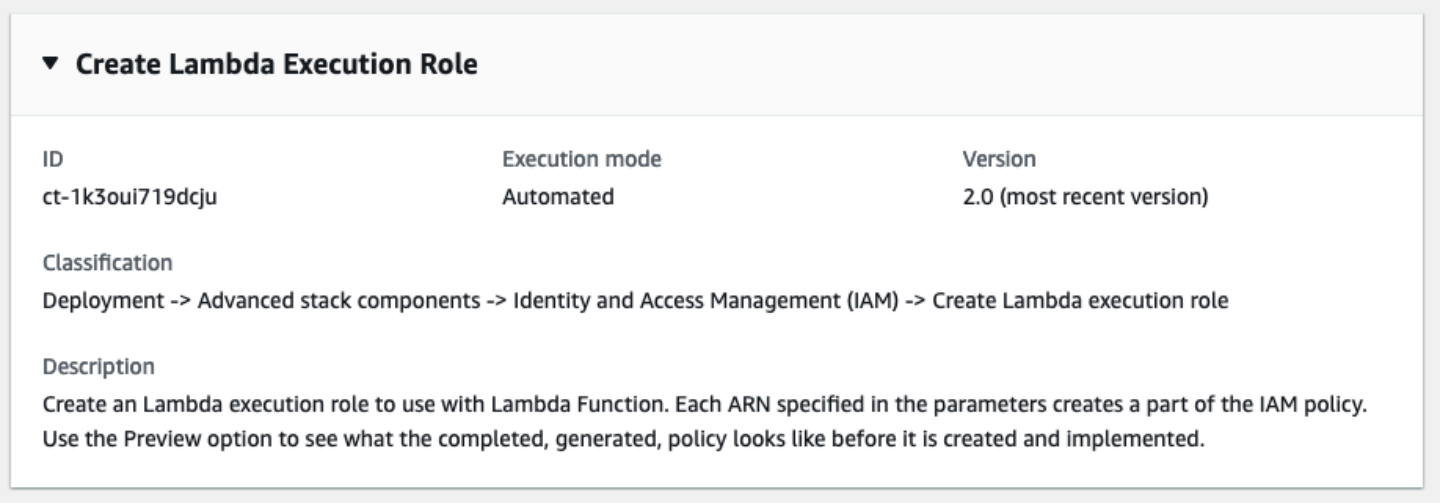
Change Type Details

Change type ID	ct-1k3oui719dcju
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM Lambda execution role

Creating IAM Lambda execution roles with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM Lambda execution roles with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE (required parameters only):

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1k3oui719dcju" --change-type-version "2.0"
 --title "Create IAM Lambda Execution Role" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleCreateIAMRole-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"ServicePrincipal\":[\"lambda.amazonaws.com\"],\"RoleName\":[\"test-
application-ec2-instance-profile\"],\"LambdaFunctionArns": [\"arn:aws:lambda:us-
east-1:123456789012:function:testing\"]}"
```

TEMPLATE CREATE (all parameters):

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamLambdaExeRoleParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1k3oui719dcju"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamLambdaExeRoleParams.json
   ```

2. Modify and save the CreateIamLambdaExeRoleParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-HandleCreateIAMRole-Admin",
  "Region": "us-east-1",
  "Parameters": {
     "ServicePrincipal" : "lambda.amazonaws.com",
     "RoleName" : "customer_lambda_execution_role",
     "VPCAccess" : "No",
     "Preview" : "No",
     "LambdaFunctionArns": ["arn:aws:lambda:us-east-1:123456789012:function:dabba"]
  }
}
```

3. Output the RFC template JSON file to a file named CreateIamLambdaExeRoleRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamLambdaExeRoleRfc.json
```

4. Modify and save the CreateIamLambdaExeRoleRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "2.0",
"ChangeTypeId": "ct-1k3oui719dcju",
"Title": "Create IAM Lambda Execution Role"
}
```

5. Create the RFC, specifying the CreateIamLambdaExeRoleRfc file and the CreateIamLambdaExeRoleParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamLambdaExeRoleRfc.json  --
execution-parameters file://CreateIamLambdaExeRoleParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 2.0 with improvements in the create RFC Console experience and the change makes it easier to copy and paste JSON.

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1k3oui719dcju.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateIAMRole-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "ServicePrincipal" : "lambda.amazonaws.com",
    "RoleName" : "customer_lambda_execution_role",
    "VPCAccess" : "No",
    "Preview" : "No",
    "LambdaFunctionArns": ["arn:aws:lambda:us-east-1:083904590739:function:dabba"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateIAMRole-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "ServicePrincipal": "lambda.amazonaws.com",
    "RoleName" : "customer_lambda_execution_role",
    "RolePath": "/test/",
    "Preview": "No",
    "LambdaFunctionArns": ["arn:aws:lambda:us-east-1:083904590739:function:dabba"],
    "VPCAccess": "Yes",
    "CloudWatchAlarmReadAccess": ["arn:aws:cloudwatch:us-
east-1:123456789012:alarm:myalarm*"],
    "CloudWatchAlarmWriteAccess": ["arn:aws:cloudwatch:us-
east-1:123456789012:alarm:myalarm*"],
    "CloudWatchLogsReadAccess": ["arn:aws:logs:us-east-1:123456789012:log-
group:myparam*:log-stream:mylogstream"],
    "CloudWatchLogsWriteAccess": ["arn:aws:logs:us-east-1:123456789012:log-
group:mylogs*"],
```

```
    "CloudWatchMetricsReadAccess": ["*"],
    "CloudWatchMetricsWriteAccess": ["Company/AppMetric"],
    "DynamoDBDataReadWriteAccess": ["arn:aws:dynamodb:us-east-1:123456789012:table/
mytable*"],
    "DynamoDBResourceReadAccess": ["arn:aws:dynamodb:us-east-1:123456789012:table/
anotherTable"],
    "KMSCryptographicOperationAccess": ["arn:aws:kms:us-
east-1:123456789012:key/97f43232-6bdc-4830-b54c-2d2926ba69aa"],
    "KMSReadAccess": ["arn:aws:kms:us-east-1:123456789012:key/97f43232-6bdc-4830-
b54c-2d2926ba69aa"],
    "S3ReadAccess": ["arn:aws:s3:::my-s3-us-east-1/*"],
    "S3WriteAccess": ["arn:aws:s3:::my-s3-ap-southeast-2/developers/design_info.doc"],
    "SNSReadAccess": ["arn:aws:sns:us-east-1:123456789012:mytopic*"],
    "SNSWriteAccess": ["arn:aws:sns:us-east-1:123456789012:MyTopic*"],
    "SQSReadAccess": ["arn:aws:sqs:us-east-1:123456789012:Myqueue*"],
    "SQSWriteAccess": ["arn:aws:sqs:us-east-1:123456789012:MyQueeu*"],
    "SSMReadAccess": ["arn:aws:ssm:us-east-1:123456789012:parameter/myparam*"],
    "SSMWriteAccess": ["arn:aws:ssm:us-east-1:123456789012:parameter/myparam*"],
    "LambdaReadAccess" : ["arn:aws:lambda:us-east-1:083904590739:function:dabba"],
    "LambdaInvokeAccess" : ["arn:aws:lambda:us-east-1:083904590739:function:dabba"],
    "EventsReadAccess" : ["arn:aws:events:us-east-1:083904590739:rule/rule01"],
    "EventsWriteAccess" : ["arn:aws:events:us-east-1:083904590739:event-bus/bus01"],
    "STSAssumeRole": ["arn:aws:iam::123456789012:role/roleName"],
    "SecretsManagerReadAccess": ["arn:aws:secretsmanager:us-
east-1:123456789012:secret:mysecret*"],
    "AdditionalPolicy" : "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Action\":[\"iam:ListRoles\",\"iam:ListAccountAliases\"],\"Resource\":\"*
\"}]}"
  }
}
```

Identity and Access Management (IAM) | Create OpenID Connect Provider

Create an IAM OpenID Connect provider for the Amazon Elastic Kubernetes Service (Amazon EKS) cluster.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create OpenID Connect provider

Change Type Details

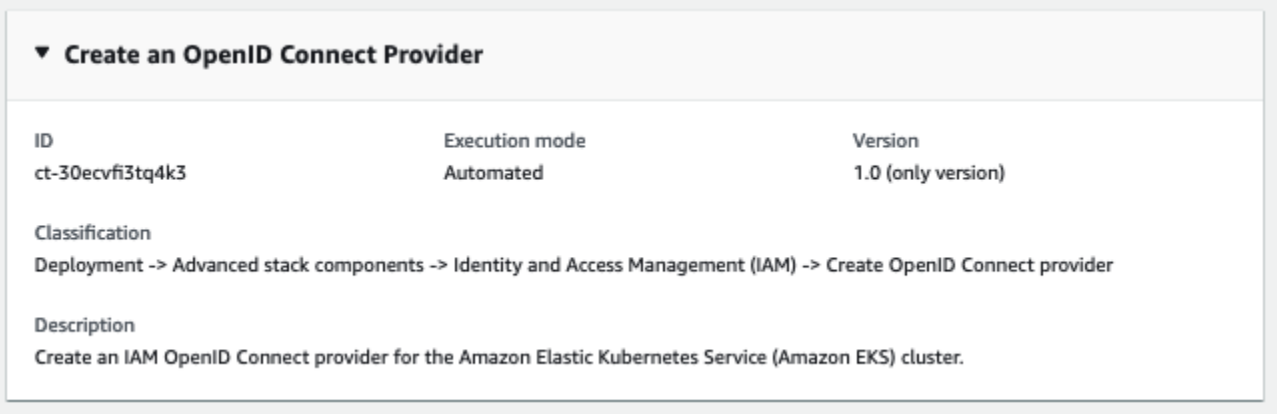
Change type ID	ct-30ecvfi3tq4k3

Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM OpenID Connect provider

Creating an IAM OpenID Connect provider with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an IAM OpenID Connect provider with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Note

When pasting in a policy document, note that the RFC only accepts policy pastes up to 5,000 characters. If your file has more than 5,000 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-30ecvfi3tq4k3" --change-type-version "1.0"
 --title "Create OpenID Connect provider" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-HandleAssociateIAMOpenIDProvider-Admin\",\"Region\": \"us-
east-1\",\"Parameters\": {\"ClusterName\": [\"test-cluster\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamOpenIdParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-30ecvfi3tq4k3"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamOpenIdParams.json
   ```

2. Modify and save the CreateIamOpenIdParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName": "AWSManagedServices-HandleAssociateIAMOpenIDProvider-Admin",
     "Region": "us-east-1",
   ```

```
  "Parameters": {
    "ClusterName": [
      "test-cluster"
    ]
  }
}
```

3. Output the RFC template JSON file to a file named CreateIamOpenIdRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamOpenIdRfc.json
```

4. Modify and save the CreateIamOpenIdRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-30ecvfi3tq4k3",
  "Title": "Create OpenID Connect provider"
}
```

5. Create the RFC, specifying the CreateIamOpenIdRfc file and the CreateIamOpenIdParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamOpenIdRfc.json  --execution-
parameters file://CreateIamOpenIdParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-30ecvfi3tq4k3.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAssociateIAMOpenIDProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "ClusterName" : [
      "test-cluster"
    ]
  }
}
```

Identity and Access Management (IAM) | Create SAML Identity Provider

Create an IAM identity provider using the SAML metadata document file that you stored in your chosen S3 bucket.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create SAML identity provider

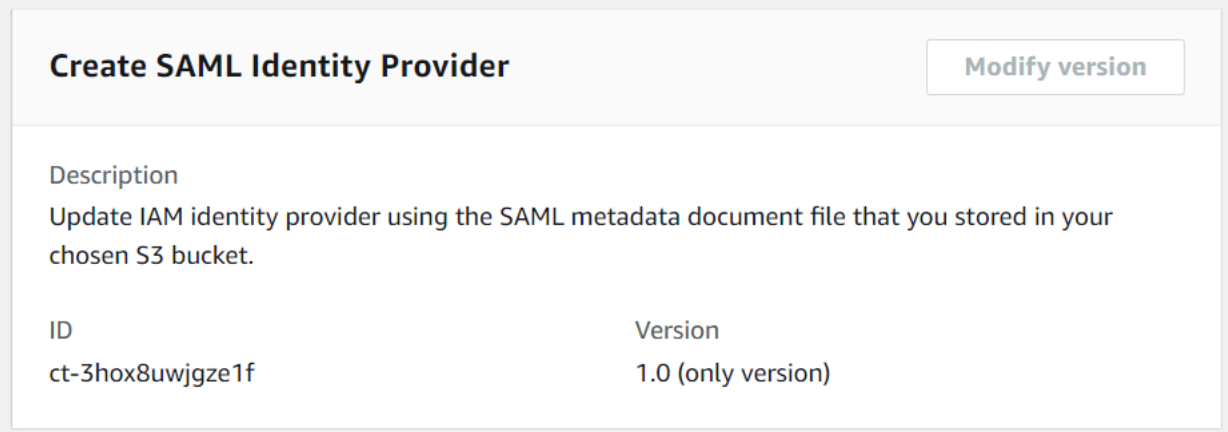
Change Type Details

Change type ID	ct-3hox8uwjgze1f
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM SAML identity provider

Creating IAM SAML IDPs with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM SAML IDPs with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3hox8uwjgze1f" --change-type-version "1.0"
 --title "Create SAML Identity Provider" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleCreateSamlProvider-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"SAMLMetadataDocumentURL\":[\"s3://bucket.name/idp-metadata.xml\"],
\"Name\":[\"customer-saml\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamSamlIdpParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3hox8uwjgze1f"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateIamSamlIdpParams.json
   ```

2. Modify and save the CreateIamSamlIdpParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName" : "AWSManagedServices-HandleCreateSamlProvider-Admin",
     "Region" : "us-east-1",
     "Parameters" : {
       "SAMLMetadataDocumentURL" : [
         "s3://bucket.name/idp-metadata.xml"
       ],
       "Name" : [
         "customer-saml"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file named CreateIamSamlIdpRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateIamSamlIdpRfc.json
   ```

4. Modify and save the CreateIamSamlIdpRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-3hox8uwjgze1f",
"Title": "Create IAM SAML identity provider"
}
```

5. Create the RFC, specifying the CreateIamSamlIdpRfc file and the CreateIamSamlIdpParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamSamlIdpRfc.json  --execution-
parameters file://CreateIamSamlIdpParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3hox8uwjgze1f.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SAMLMetadataDocumentURL" : [
      "s3://mybucket/path/to/metadata.xml"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SAMLMetadataDocumentURL" : [
      "s3://mybucket/path/to/metadata.xml"
    ],
    "Name" : [
      "customer-saml"
    ]
  }
}
```

Identity and Access Management (IAM) | Create Service-Linked Role

Create an IAM service-linked role linked to an AWS service that you specify.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Service-Linked role

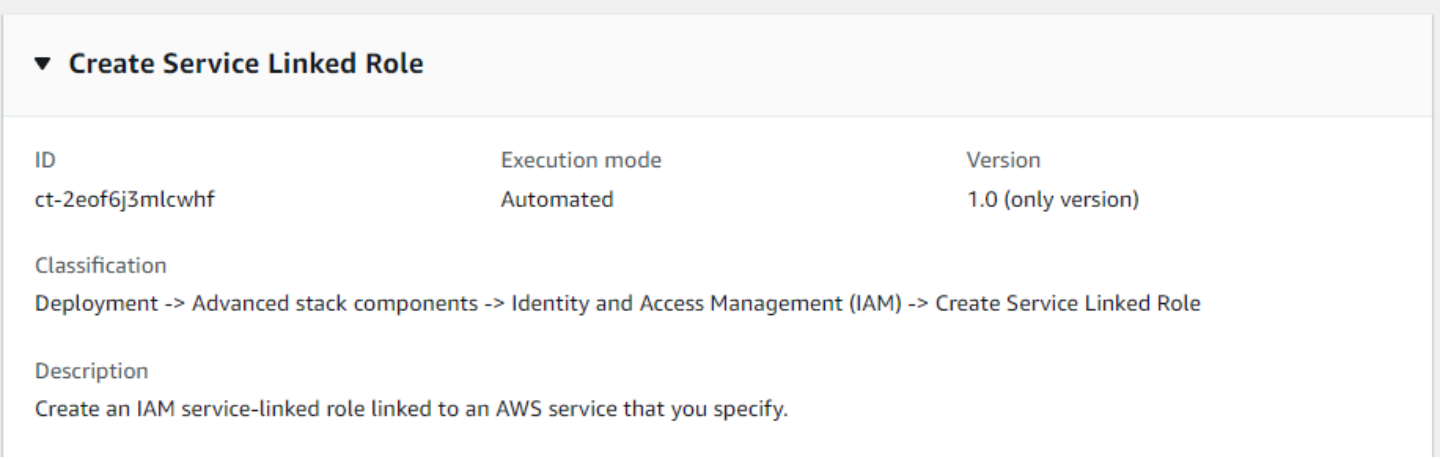
Change Type Details

Change type ID	ct-2eof6j3mlcwhf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create IAM service-linked role

Creating IAM service-linked roles with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM service-linked roles with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

> **Note**
>
> When pasting in a policy document, note that the RFC only accepts policy pastes up to 5,000 characters. If your file has more than 5,000 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2eof6j3mlcwhf" --change-type-version "1.0"
 --title "Create service-linked role" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-CreateServiceLinkedRole-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"AWSServiceName\": [\"acm.amazonaws.com\"],\"Description\":
 [\"AWSServiceRoleForCertificateManager\"]}}"
```

```
aws amscm create-rfc --change-type-id "ct-2eof6j3mlcwhf" --change-type-version "1.0"
 --title "Create service-linked role" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-CreateServiceLinkedRole-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"AWSServiceName\": [\"acm.amazonaws.com\"],\"Description\":
 [\"AWSServiceRoleForCertificateManager\",\"CustomSuffix\": [\"CustomSuffix-Dev\"]}}"
```

TEMPLATE CREATE:

1. Save a CreateSlrRfc.json file.

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-2eof6j3mlcwhf",
     "Title": "Create service-linked role"
   }
   ```

2. Save a CreateSlrParams.json file. For example, you can replace the contents with something like this:

   ```
   {
   "DocumentName": "AWSManagedServices-CreateServiceLinkedRole-Admin",
   ```

```
"Region": "us-east-1",
"Parameters": {
 "AWSServiceName": [ "acm.amazonaws.com" ],
 "Description" : ["AWSServiceRoleForCertificateManager" ],
 "CustomSuffix" : ["CustomSuffix-Dev" ]
}
}
```

3. Create the RFC, specifying the CreateSlrRfc file and the CreateSlrParams files:

```
aws amscm create-rfc --cli-input-json file://CreateSlrRfc.json --execution-
parameters file://CreateSlrParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

For more information about service-linked roles, see Using service-linked roles.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2eof6j3mlcwhf.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateServiceLinkedRole-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "AWSServiceName" : [
      "autoscaling.amazonaws.com"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateServiceLinkedRole-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "AWSServiceName": [
      "autoscaling.amazonaws.com"
    ],
    "CustomSuffix": [
      "test123"
    ],
    "Description": [
      ""
    ]
  }
}
```

Identity and Access Management (IAM) | Create Service-Specific Credentials

Generate a set of credentials consisting of a user name and password, to use to access the specified service.

Full classification: Deployment | Advanced stack components | Identity and Access Management (IAM) | Create service-specific credentials

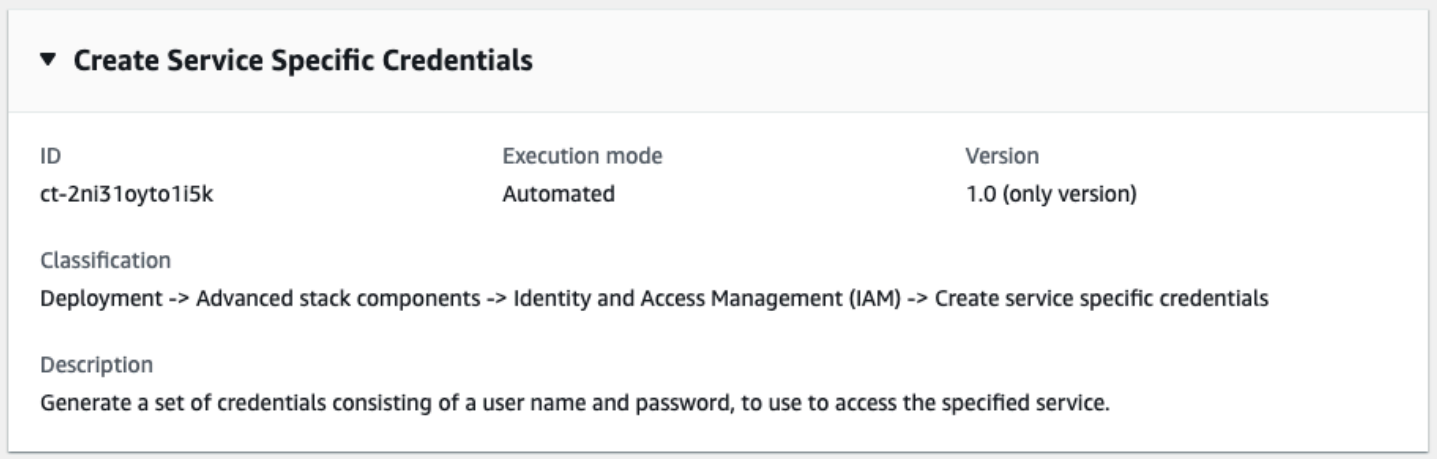
Change Type Details

Change type ID	ct-2ni31oyto1i5k
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create service specific credentials

Creating IAM service specific credentials with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM service specific credentials with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-2ni31oyto1i5k" \
--change-type-version "1.0" --title "Create service specific credentials for IAM User"
 \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
CreateServiceSpecificCredentials\",\"Region\":\"us-east-1\",\"Parameters\":{\"Username
\":[\"testuser\"],\"Service\":[\"CodeCommit\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateServSpecCredsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2ni31oyto1i5k"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateServSpecCredsParams.json
   ```

2. Modify and save the CreateServSpecCredsParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName" : "AWSManagedServices-CreateServiceSpecificCredentials",
     "Region" : "us-east-1",
     "Parameters" : {
       "Username" : [
         "testuser"
       ],
       "Service" : [
         "CodeCommit"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file named CreateServSpecCredsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateServSpecCredsRfc.json
```

4. Modify and save the CreateServSpecCredsRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-2ni31oyto1i5k",
  "ChangeTypeVersion": "1.0",
  "Title": "Testing ct-2ni31oyto1i5k CreateServiceSpecificCredentials in region us-east-1 for an IAM User"
}
```

5. Create the RFC, specifying the CreateServSpecCredsRfc file and the CreateServSpecCredsParams file:

```
aws amscm create-rfc --cli-input-json file://CreateServSpecCredsRfc.json  --
execution-parameters file://CreateServSpecCredsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2ni31oyto1i5k.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateServiceSpecificCredentials",
  "Region" : "us-east-1",
  "Parameters" : {
    "Username" : [
      "testuser"
    ],
```

```
    "Service" : [
      "CodeCommit"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateServiceSpecificCredentials",
  "Region" : "us-east-1",
  "Parameters" : {
    "Username" : [
      "testuser"
    ],
    "Service" : [
      "CodeCommit"
    ]
  }
}
```

KMS Alias | Create

Create an alias for an AWS Key Management Service (KMS) customer master key (CMK).

Full classification: Deployment | Advanced stack components | KMS alias | Create

Change Type Details

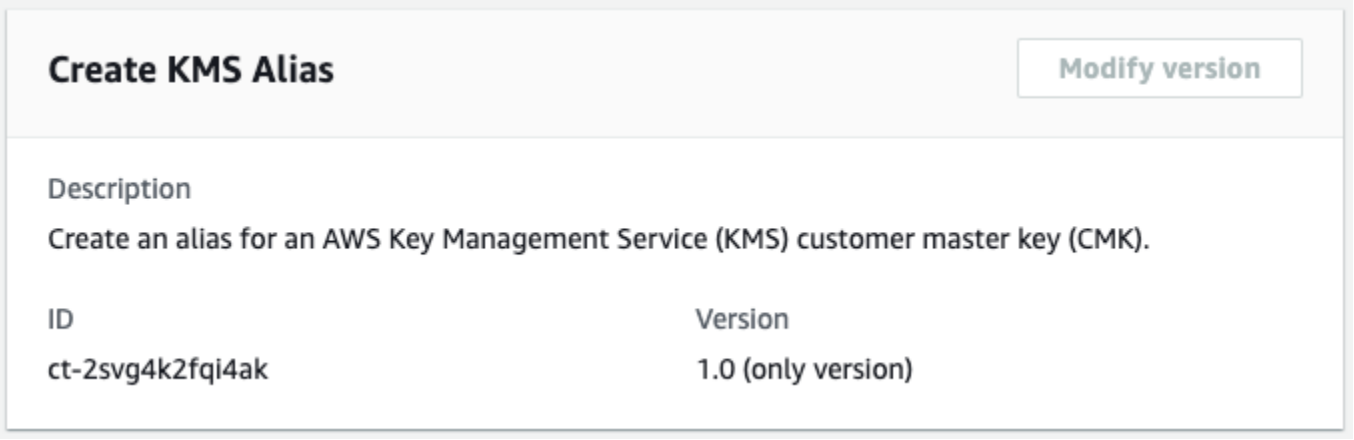
Change type ID	ct-2svg4k2fqi4ak
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create an AWS KMS alias

Creating an AWS KMS alias with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AWS KMS alias with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title create-kms-alias --change-type-id ct-2svg4k2fqi4ak --
change-type-version 1.0 --execution-parameters '{"DocumentName": "AWSManagedServices-
CreateKMSAlias", "Region": "us-east-1", "Parameters": {"TargetKeyId": ["12345678-90ab-
cdef-1234-567890abcdef"], "AliasName": ["my-test-key"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateKmsAliasParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2svg4k2fqi4ak" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateKmsAliasParams.json
   ```

2. Modify and save the CreateKmsAliasParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateKMSAlias",
     "Region": "us-east-1",
     "Parameters": {
       "TargetKeyId": ["12345678-90ab-cdef-1234-567890abcdef"]
       "AliasName": ["my-test-key"]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it CreateKmsAliasRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateKmsAliasRfc.json
   ```

4. Modify and save the CreateKmsAliasRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-2svg4k2fqi4ak",
   ```

```
"Title":                "create-kms-alias"
}
```

5. Create the RFC, specifying the CreateKmsAlias Rfc file and the CreateKmsAliasParams file:

```
aws amscm create-rfc --cli-input-json file://CreateKmsAliasRfc.json  --execution-
parameters file://CreateKmsAliasParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2svg4k2fqi4ak.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateKMSAlias",
  "Region" : "us-east-1",
  "Parameters" : {
    "TargetKeyId" : [
      "58c399bf-1662-4d55-8bbe-fb6d26bd72b9"
    ],
    "AliasName" : [
      "test-alias"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateKMSAlias",
  "Region" : "us-east-1",
  "Parameters" : {
    "TargetKeyId" : [
      "arn:aws:kms:us-east-1:123456789012:key/58c399bf-1662-4d55-8bbe-fb6d26bd72b9"
```

```
    ],
    "AliasName" : [
      "test-alias"
    ]
  }
}
```

KMS Key | Create

Request a KMS key with a predefined key policy.

Full classification: Deployment | Advanced stack components | KMS key | Create

Change Type Details

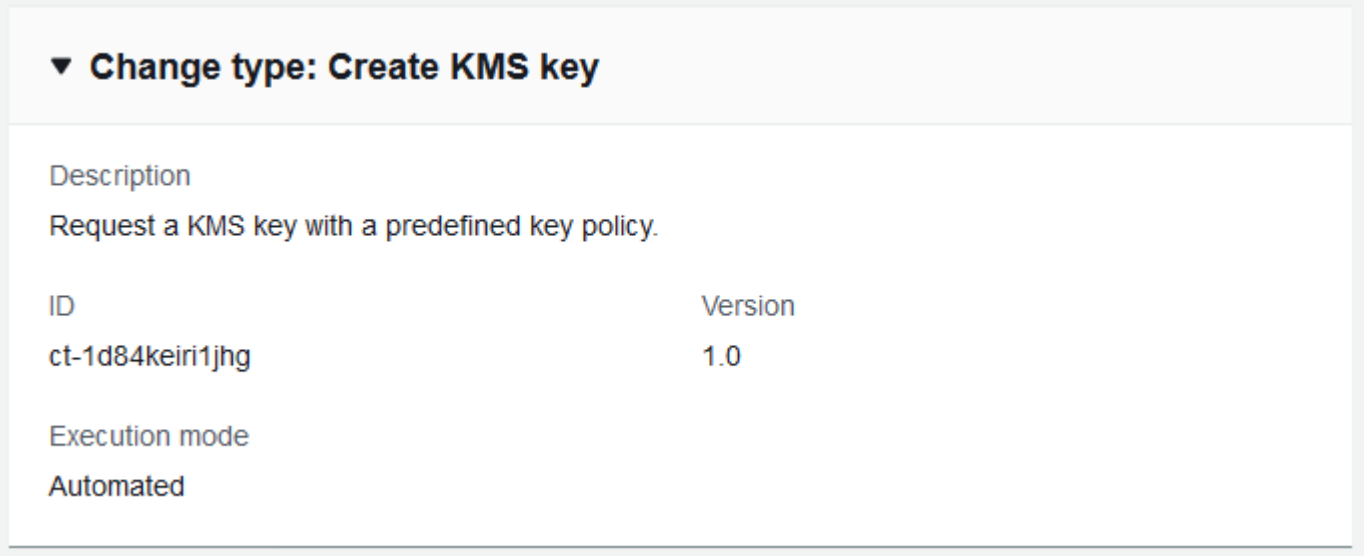
Change type ID	ct-1d84keiri1jhg
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create KMS key

Creating an AWS KMS Key with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AWS KMS Key with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Required parameters only:

```
aws amscm create-rfc --title my-app-key --change-type-id ct-1d84keiri1jhg
 --change-type-version 1.0 --execution-parameters '{"Description":"KMS key
 for my-app","VpcId":"VPC_ID","Name":"my-app-key","StackTemplateId":"stm-
enf1j068fhg34vugt","TimeoutInMinutes":60,"Parameters":{"Description":"KMS key for my-
app"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateKmsKeyAutoParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1d84keiri1jhg"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateKmsKeyAutoParams.json
   ```

2. Modify and save the CreateKmsKeyAutoParams file. Examples follow.

   Grant a user or a role, permission to decrypt the created CMK. Example execution parameters:

   ```
   {
     "Description": "KMS key for my-app",
     "VpcId": "VPC_ID”,
     "Name": "my-app-key-decrypt",
     "StackTemplateId": "stm-enf1j068fhg34vugt",
     "TimeoutInMinutes": 60,
     "Parameters": {
       "IAMPrincipalsRequiringDecryptPermissions": [
         "ARN:role/roleA",
         "ARN:user/userB",
         "ARN:role/instanceProfileA"
       ],
       "Description": "KMS key for my-app"
     }
   }
   ```

For the resulting policy, see Grants Permissions to Decrypt with the CML for an IAM User or a Role.

Grant a user or role, permission to encrypt using the created CMK. Example execution parameters:

```
{
  "Description": "KMS key for my-app",
  "VpcId": "VPC_ID",
  "Name": "my-app-key-encrypt",
  "Tags": [
    {
      "Key": "Name",
      "Value": "my-app-key"
    }
  ],
  "StackTemplateId": "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "IAMPrincipalsRequiringEncryptPermissions": [
      "ARN:role/roleA",
      "ARN:user/userB",
      "ARN:role/instanceProfileA"
    ],
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Grants Permissions to Encrypt with the CML for an IAM User or a Role.

Grant a user, role, or account, permission to create grants using the created CMK. Example execution parameters:

```
{
  "Description": "KMS key for my-app",
  "VpcId": "VPC_ID",
  "Name": "my-app-key-create-grants",
  "StackTemplateId": "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes": 60,
  "Parameters": {
```

```
    "IAMPrincipalsRequiringGrantsPermissions": [
      "arn:aws:iam::999999999999:role/roleA",
      "888888888888"
    ],
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Grants Permissions to Create Grants with the CMK for an IAM User, Role, or Account.

Allow only AWS services that are integrated with AWS KMS to perform the GRANT operation. Example execution parameters:

```
{
  "Description": "KMS key for my-app",
  "VpcId": "VPC_ID",
  "Name": "my-app-key-limit-to-services",
  "StackTemplateId": "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "IAMPrincipalsRequiringGrantsPermissions": [
      "arn:aws:iam::999999999999:role/roleA"
    ],
    "LimitGrantsToAWSResources": "true",
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Allow only AWS services that are Integrated with AWS KMS to Perform the GRANT Operation.

Enforce use of encryption context keys in cryptographic operations. Example execution parameters:

```
{
  "Description": "KMS key for my-app",
  "VpcId": "VPC_ID",
  "Name": "my-app-key-encryption-keys",
  "StackTemplateId": "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes": 60,
  "Parameters": {
```

```
    "EnforceEncryptionContextKeys": "true",
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Enforce use of Encryption Context Keys in Cryptographic Operations.

Enforce a specific list of encryption context keys in cryptographic operations. Example execution parameters:

```
{
  "Description": "KMS key for my-app",
  "VpcId": "VPC_ID",
  "Name": "my-app-key-encryption-list",
  "StackTemplateId": "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "AllowedEncryptionContextKeys": [
      "Name",
      "Application"
    ],
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Enforce a Specific List of Encryption Context Keys in Cryptographic Operations.

Allow AWS services to access the created CMK. Example execution parameters:

```
{
  "Description" : "KMS key for my-app",
  "VpcId" : "VPC_ID",
  "Name" : "my-app-key-allow-aws-service-access",
  "StackTemplateId" : "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
      "AllowServiceRolesAccessKMSKeys": [
          "ec2.us-east-1.amazonaws.com",
          "ecr.us-east-1.amazonaws.com"
    ],
```

```
    "Description": "KMS key for my-app"
  }
}
```

For the resulting policy, see Allow AWS services to access created CMK.

3. Output the RFC template JSON file to a file; this example names it CreateKmsKeyAutoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateKmsKeyAutoRfc.json
```

4. Modify and save the CreateKmsKeyAutoRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-1d84keiri1jhg",
    "ChangeTypeVersion": "1.0",
    "Title": "Create KMS Key"
}
```

5. Create the RFC, specifying the CreateKmsKeyAuto Rfc file and the CreateKmsKeyAutoParams file:

```
aws amscm create-rfc --cli-input-json file://CreateKmsKeyAutoRfc.json  --execution-
parameters file://CreateKmsKeyAutoParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- This CT creates a CloudFormation stack which creates a KMS key with `DeletionPolicy: Retain`. By design, the created KMS key will persist even after you delete the stack. If you are sure you want to delete the KMS key, create an RFC with Change Type ct-2zxya20wmf5bf, Management | Advanced stack components | KMS key | Delete (review required).
- This change type is ExecutionMode=Automated, so this change type does not require manual review by AMS operations and should execute more rapidly than KMS Key: Create (review required); however, if you have an unusual situation, the manual version might work better for you. See KMS Key | Create (Review Required).

- This CT has a new parameter, AllowServiceRolesAccessKMSKeys, that provides the specified AWS services access to the KMS key. The change was made because the Autoscaling group service role was unable to start the EC2 instances with encrypted EBS volumes due to lack of permissions to the KMS key.
- To learn more about AWS KMS keys, see AWS Key Management Service (KMS), AWS Key Management Service FAQs, and AWS Key Management Service Concepts.

KMS Key Create resulting policies

Depending on how you created your KMS key, you created policies. These example policies match various KMS key create scenarios provided in Create KMS key.

Grants Permissions to Decrypt with the CML for an IAM User or a Role

Resulting example policy grants IAM users, roles or instance profiles, permission to decrypt using the CMK:

```
{
          "Sid": "Allow decrypt using the key",
          "Effect": "Allow",
          "Principal": {
              "AWS": [
                "arn:aws:iam::999999999999:role/roleA",
                "arn:aws:iam::999999999999:user/userB",
                "arn:aws:iam::999999999999:role/instanceProfileA"
              ]
          },
          "Action": [
              "kms:DescribeKey",
              "kms:Decrypt"
          ],
          "Resource": "*"
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Grants Permissions to Encrypt with the CML for an IAM User or a Role

Resulting example policy grants IAM users, roles or instance profiles, permission to encrypt using the CMK:

```
{
          "Sid": "Allow encrypt using the key",
          "Effect": "Allow",
          "Principal": {
              "AWS": [
                "arn:aws:iam::999999999999:role/roleA",
                "arn:aws:iam::999999999999:user/userB",
                "arn:aws:iam::999999999999:role/instanceProfileA"
              ]
          },
          "Action": [
              "kms:DescribeKey",
              "kms:Encrypt",
              "kms:ReEncrypt*",
              "kms:GenerateDataKey",
              "kms:GenerateDataKeyWithoutPlaintext"
          ],
          "Resource": "*"
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Grants Permissions to Create Grants with the CMK for an IAM User, Role, or Account

Resulting example policy:

```
{
          "Sid": "Allow grants",
          "Effect": "Allow",
          "Principal": {
              "AWS": [
                  "arn:aws:iam::999999999999:role/roleA",
                  "arn:aws:iam::888888888888:root"
              ]
          },
          "Action": [
              "kms:CreateGrant",
              "kms:ListGrants",
              "kms:RevokeGrant"
          ],
          "Resource": "*"
```

```
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Allow only AWS services that are Integrated with AWS KMS to Perform the GRANT Operation

Resulting example policy:

```
{
          "Sid": "Allow grants",
          "Effect": "Allow",
          "Principal": {
              "AWS": "arn:aws:iam::999999999999:role/roleA"
          },
          "Action": [
              "kms:CreateGrant",
              "kms:ListGrants",
              "kms:RevokeGrant"
          ],
          "Resource": "*"
      },
      {
          "Sid": "Deny if grant is not for AWS resource",
          "Effect": "Deny",
          "Principal": {
              "AWS": "*"
          },
          "Action": [
              "kms:CreateGrant",
              "kms:ListGrants",
              "kms:RevokeGrant"
          ],
          "Resource": "*",
          "Condition": {
              "Bool": {
                  "kms:GrantIsForAWSResource": "false"
              }
          }
      } }
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Enforce use of Encryption Context Keys in Cryptographic Operations

Resulting example policy:

```
{
          "Effect": "Deny",
          "Principal": {
              "AWS": "*"
          },
          "Action": [
              "kms:CreateGrant",
              "kms:Decrypt",
              "kms:Encrypt",
              "kms:GenerateDataKey*",
              "kms:ReEncrypt"
          ],
          "Resource": "*",
          "Condition": {
              "Null": {
                  "kms:EncryptionContextKeys": "true"
              }
          }
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Enforce a Specific List of Encryption Context Keys in Cryptographic Operations

Resulting example policy:

```
{
          "Effect": "Deny",
          "Principal": {
              "AWS": "*"
          },
          "Action": [
              "kms:CreateGrant",
              "kms:Decrypt",
```

```
            "kms:Encrypt",
            "kms:GenerateDataKey*",
            "kms:ReEncrypt"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "kms:EncryptionContextKeys": [
                  "Name",
                  "Application"
                ]
            }
        }
    }
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Allow AWS services to access created CMK

Resulting example policy:

```
{
    "Effect": "Allow",
    "Principal": {
        "AWS": "*"
    },
    "Action": [
        "kms:ListGrants",
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:ViaService":  [
              "ec2.us-west-2.amazonaws.com",
              "ecr.us-east-1.amazonaws.com"
            ],
```

```
            "kms:CallerAccount": "111122223333"
        }
    }
}
```

For the execution parameters to create this policy with the KMS key Create change type, see Create KMS key

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1d84keiri1jhg.

Example: Required Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "Description" : "Test key"
  }
}
```

Example: All Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678",
  "Name" : "TestStack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-enf1j068fhg34vugt",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "Alias" : "testkey",
```

```
    "EnableKeyRotation" : "true",
    "Description" : "Test key",
    "PendingWindow" : 30,
    "IAMPrincipalsRequiringDecryptPermissions" : [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "IAMPrincipalsRequiringEncryptPermissions" : [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "IAMPrincipalsRequiringGrantsPermissions" : [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole",
      "987654321098"
    ],
    "LimitGrantsToAWSResources" : "true",
    "EnforceEncryptionContextKeys" : "true",
    "AllowedEncryptionContextKeys" : [
      "App"
    ],
    "AllowServiceRolesAccessKMSKeys": [
      "ec2.us-east-1.amazonaws.com"
    ]
  }
}
```

KMS Key | Create (Review Required)

Request a KMS key by describing key permissions or submitting a key policy document.

Full classification: Deployment | Advanced stack components | KMS key | Create (review required)

Change Type Details

Change type ID	ct-2epp05svrlwod
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required

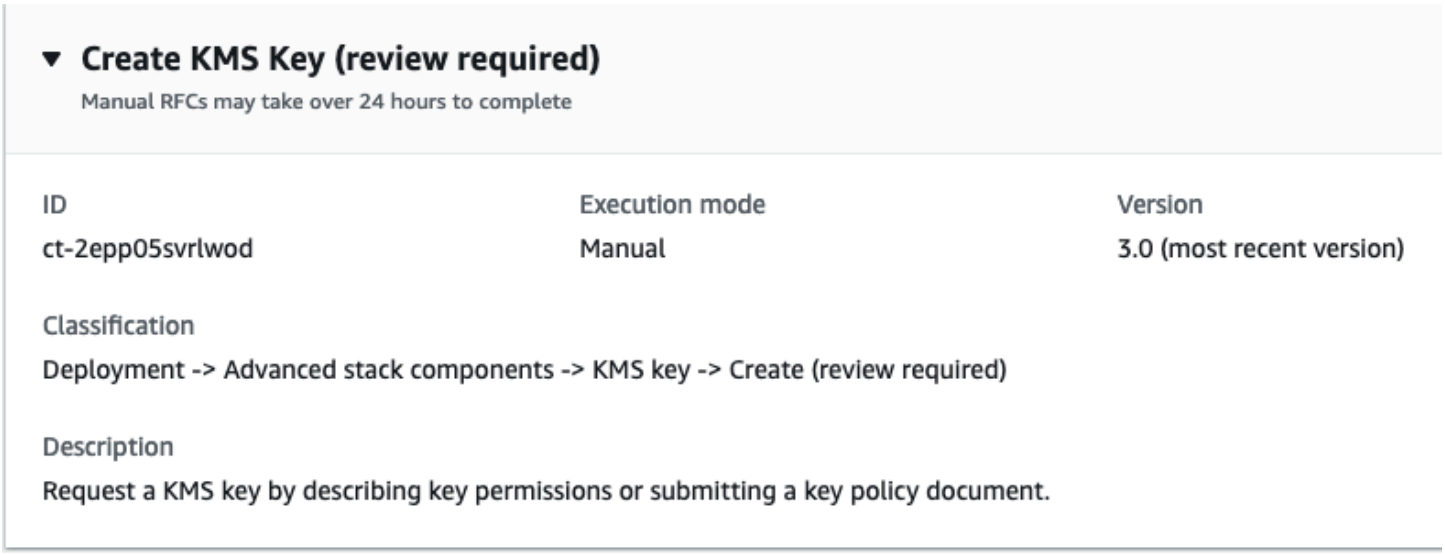
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create KMS Key (review required)

Creating an AWS KMS Key (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AWS KMS Key (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2epp05svrlwod" --change-type-version "3.0"
 --title "TITLE" --execution-parameters "{\"KeyDescription\": \"Example description\",
 \"KeyPermissions\": \"key permissions\", \"Operation\": \"Create\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateKmsKeyParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2epp05svrlwod" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateKmsKeyParams.json
   ```

2. Modify and save the CreateKmsKeyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "KMS key request",
     "KeyPermissions": "{\"Id\":\"key-consolepolicy-3\",\"Version\":\"2012-10-17\",
   \"Statement\":[{\"Sid\":\"Allow use of the key\",\"Effect\":\"Allow\",\"Principal
   \":{\"AWS\":[\"arn:aws:iam::111122223333:role/KMSRole\"]},\"Action\":[\"kms:Encrypt
   \",\"kms:Decrypt\",\"kms:ReEncrypt*\",\"kms:GenerateDataKey*\",\"kms:DescribeKey
   \"],\"Resource\":\"*\"}]}",
     "Operation": "Create"
   }
   ```

3. Output the RFC template JSON file to a file; this example names it CreateKmsKeyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateKmsKeyRfc.json
   ```

4. Modify and save the CreateKmsKeyRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-2epp05svrlwod",
"Title":                "KmsKey-Create-RFC"
}
```

5. Create the RFC, specifying the CreateKmsKey Rfc file and the CreateKmsKeyParams file:

```
aws amscm create-rfc --cli-input-json file://CreateKmsKeyRfc.json  --execution-
parameters file://CreateKmsKeyParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is at v3.0. The required **KeyName** parameter has been replaced by an optional **AliasName** parameter; KMS keys use aliases.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

To learn more about AWS KMS keys, see AWS Key Management Service (KMS), AWS Key Management Service FAQs, and AWS Key Management Service Concepts.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2epp05svrlwod.

Example: Required Parameters

```
{
  "KeyDescription": "Exmample description of the key to be created.",
  "KeyPermissions": "KMS Key permissions to add: kms:Get",
  "Operation": "Create"
}
```

Example: All Parameters

```
{
  "KeyDescription": "Exmample description of the key to be created.",
  "AliasName": "testkmskey",
  "KeyRotation": true,
  "KeyPermissions": "KMS Key permissions to add: kms:Get",
  "MultiRegion": false,
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "Operation": "Create",
  "Priority": "Medium"
}
```

Listener | Create (For ALB or NLB)

Create a listener for an Application Load Balancer (ALB) or Network Load Balancer (NLB). A listener is a process that checks for connection requests, the rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Full classification: Deployment | Advanced stack components | Listener | Create (for ALB or NLB)

Change Type Details

Change type ID	ct-14yjom3kvpinu

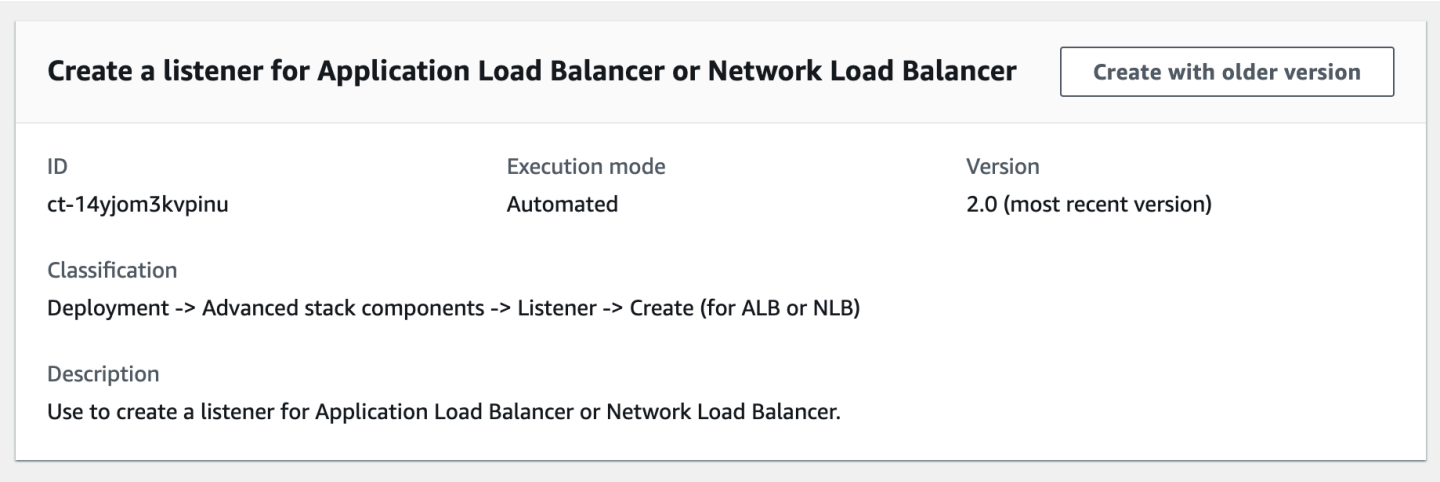
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create listener

Creating a Listener for an ALB or NLB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Listener for an ALB or NLB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-14yjom3kvpinu" --change-type-version "2.0" --title "TITLE" --execution-parameters
 "{\"Description\":\"DESCRIPTION\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\": \"stm-
u5n0r6aacdvdwthhm\", \"Name\":\"NAME\", \"TimeoutInMinutes\":60, \"Parameters\":
 {\"LoadBalancerArn\":\"LB-ARN\",\"DefaultActionTargetGroupArn\":\"TARGET-GROUP-ARN",
\"Port\":\"80\",\"Protocol\":\"HTTP\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateListenerParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-14yjom3kvpinu" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateListenerParams.json
   ```

2. Modify and save the CreateListenerParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "Listener-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-u5n0r6aacdvdwthhm",
   "Name":             "My-Listener",
   ```

```
"Parameters":   {
    "LoadBalancerArn":              ARN,
    "DefaultActionTargetGroupArn":  ARN,
    "Port":                         PORT,
    "Protocol":                    Protocol"
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateListenerRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateListenerRfc.json
```

4. Modify and save the CreateListenerRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-14yjom3kvpinu",
"Title":                "Listener-Create-RFC"
}
```

5. Create the RFC, specifying the CreateListenerRfc file and the CreateListenerParams file:

```
aws amscm create-rfc --cli-input-json file://CreateListenerRfc.json --execution-
parameters file://CreateListenerParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Next Steps: Submit a Management | Other | Other | Update change type to open ports and associate security groups, see Other | Other requests.

Tips

> **Note**
>
> You can specify up to four Target IDs, Ports, and Availability Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-14yjom3kvpinu.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "DefaultActionTargetGroupArn": "arn:aws:elasticloadbalancing:eu-
west-1:123456789012:targetgroup/target-group-name/123456789012",
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-app-load-balancer/abcdefghij",
    "Port": "80",
    "Protocol": "HTTP"
  },
  "StackTemplateId": "stm-u5n0r6aacdvdwthhm",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "CertificateArn": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
    "DefaultActionTargetGroupArn": "arn:aws:elasticloadbalancing:eu-
west-1:123456789012:targetgroup/target-group-name/123456789012",
    "LoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-app-load-balancer/abcdefghij",
    "Port": "443",
    "Protocol": "HTTP",
    "ALBSslPolicy": "ELBSecurityPolicy-2016-08",
    "AlpnPolicy": "HTTP2Only"
  },
  "StackTemplateId": "stm-u5n0r6aacdvdwthhm",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
```

```
}
```

Load Balancer (ELB) Stack | Create

Use to create an Amazon ELB Classic Load Balancer. Use alternate change types to create an Application Load Balancer (ct-111r1yayblnw4) or Network Load Balancer (ct-2qldv4h9osmau).

Full classification: Deployment | Advanced stack components | Load balancer (ELB) stack | Create

Change Type Details

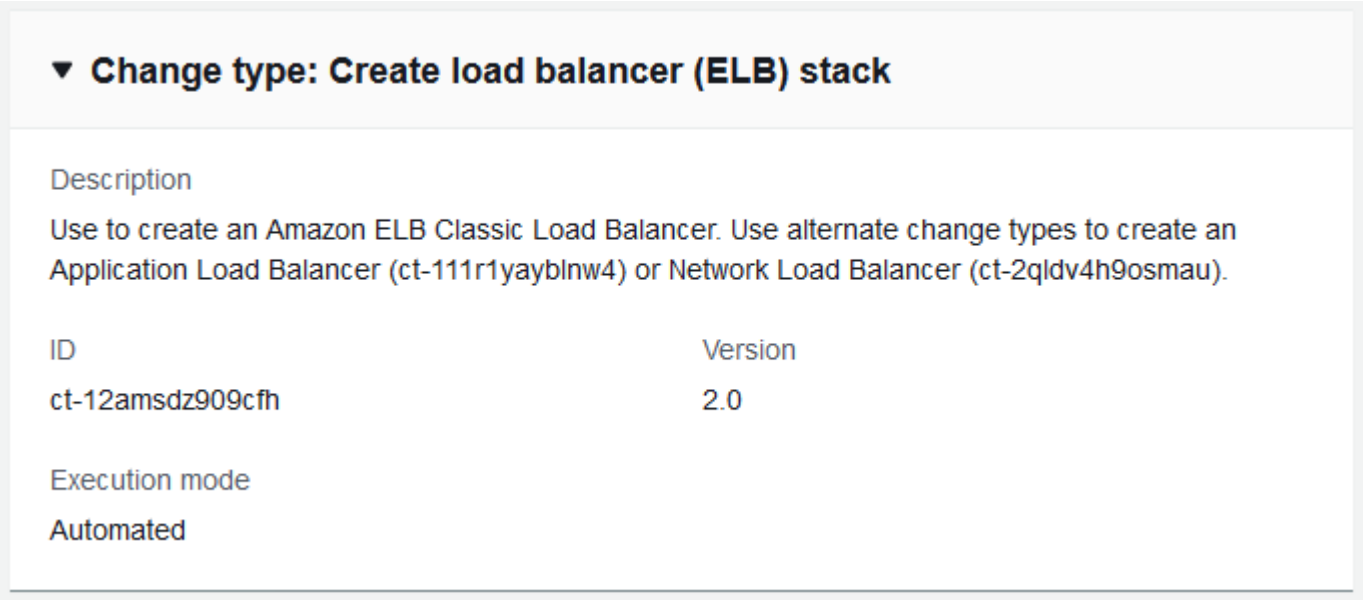
Change type ID	ct-12amsdz909cfh
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ELB load balancer

Creating an Elastic Load Balancer with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Elastic Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id ct-12amsdz909cfh --
change-type-version 3.0 --title "my-elb" --execution-parameters
 '{"Description":"My ELB","VpcId":"VPC_ID","StackTemplateId":"stm-
sdhopv30000000000","Name":"myElb","TimeoutInMinutes":60,"Parameters":{"ELBSubnetIds":
["SUBNET_ID","SUBNET_ID"],"ELBLoadBalancerPort":"80","ELBLoadBalancerProtocol":"HTTP","ELBInsta
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateElbParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-12amsdz909cfh" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateElbParams.json
   ```

2. Modify and save the CreateElbParams file. The values given in the example reflect a deployment of a Public ELB, with the health check thresholds relaxed and the ELBScheme set to `true` (for a public ELB). Note that the `Name` you set here is not the actual ELB name, you can find that name in the console as the ELB instance name. Not all optional parameters are shown in the example.

   ```
   {
   "Description":      "ELB-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-sdhopv00000000000",
   "Name":             "My-ELB",

   "Parameters":   {
       "ELBSubnetIds":  ["PUBLIC_AZ1", "PUBLIC_AZ2"],
       "ELBHealthCheckHealthyThreshold":   2,
       "ELBHealthCheckInterval":           30,
       "ELBHealthCheckTarget":             "HTTP:80/status",
       "ELBHealthCheckTimeout":            10,
       "ELBHealthCheckUnhealthyThreshold": 3,
       "ELBScheme":                        true
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it CreateElbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateElbRfc.json
```

4. Modify and save the CreateElbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-12amsdz909cfh",
"Title":                "ELB-Create-RFC"
}
```

5. Create the RFC, specifying the CreateElbRfc file and the CreateElbParams file:

```
aws amscm create-rfc --cli-input-json file://CreateElbRfc.json --execution-
parameters file://CreateElbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the load balancer, look in the execution output: Use the `stack_id` to view the ELB in the Cloud Formation console or to create a Delete Stack RFC, use the ELBCName value to programmatically access the ELB.

   You might need to submit a Management | Other | Other | Update change type to open ports and associate security groups, see Other | Other requests.

Tips

To learn more about AWS Classic Load Balancers, see What Is a Classic Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-12amsdz909cfh.

Example: Required Parameters

```
{
```

```
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-sdhopv30000000000",
  "Name": "Test Stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ELBSubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"],
    "ELBInstancePort": "80",
    "ELBInstanceProtocol": "HTTP",
    "ELBLoadBalancerPort": "80",
    "ELBLoadBalancerProtocol": "HTTP"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-sdhopv30000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ELBSubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"],
    "ELBHealthCheckHealthyThreshold": 2,
    "ELBHealthCheckInterval": 10,
    "ELBHealthCheckTarget": "HTTP:80/index.html",
    "ELBHealthCheckTimeout": 10,
    "ELBHealthCheckUnhealthyThreshold": 3,
    "ELBIdleTimeout": 30,
    "ELBInstancePort": "80",
    "ELBInstanceProtocol": "HTTPS",
    "ELBCookieExpirationPeriod": "60",
```

```
    "ELBCookieStickinessPolicyName": "MyPolicy",
    "ELBLoadBalancerName": "MyLoadBalancer",
    "ELBLoadBalancerPort": "443",
    "ELBLoadBalancerProtocol": "HTTP",
    "ELBSSLCertificateId": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
    "ELBScheme": false,
    "ELBCrossZone": true,
    "ELBBackendInstances": ["i-12345678", "i-01234567"],
    "ELBInstancePort2": "80",
    "ELBInstanceProtocol2": "HTTPS",
    "ELBCookieExpirationPeriod2": "60",
    "ELBCookieStickinessPolicyName2": "MyPolicy2",
    "ELBLoadBalancerPort2": "445",
    "ELBLoadBalancerProtocol2": "HTTP",
    "ELBSSLCertificateId2": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

Load Balancer (ELB) Stack | Create (With Additional Listeners)

Create an Elastic ("Classic") load balancer (ELB).

Full classification: Deployment | Advanced stack components | Load balancer (ELB) stack | Create (with additional listeners)

Change Type Details

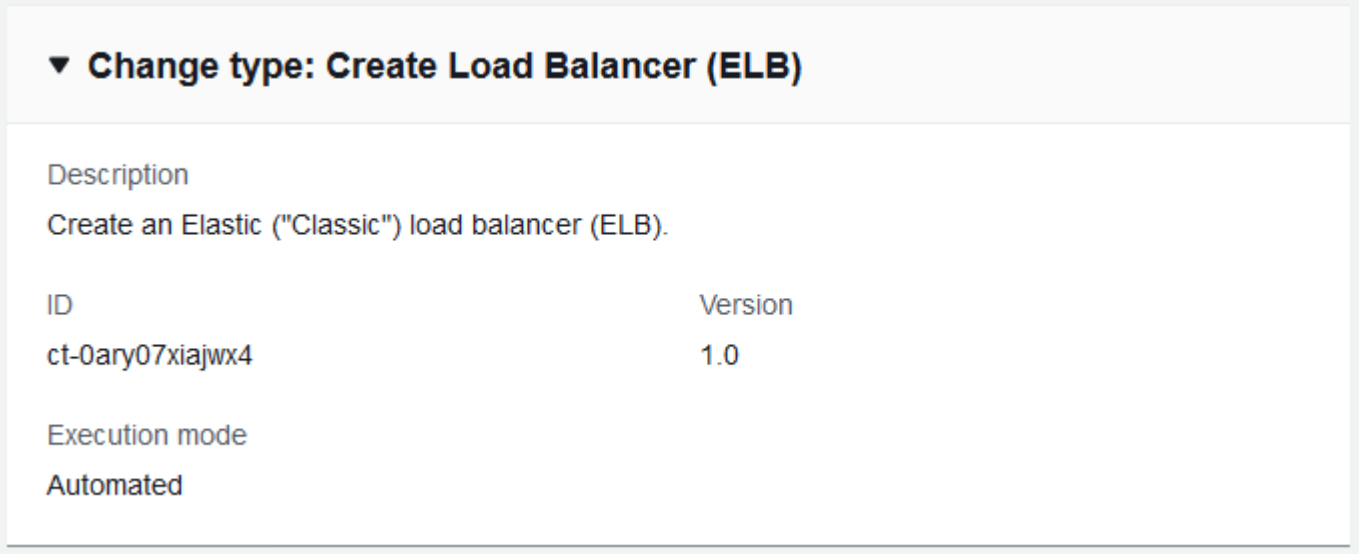
Change type ID	ct-0ary07xiajwx4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ELB load balancer with additional listeners

Creating an Elastic Load Balancer (with additional listeners) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Elastic Load Balancer (with additional listeners) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-
id "ct-0ary07xiajwx4" --change-type-version "1.0" --title 'My-ELB-AL-
Create-RFC' --description "Test"  --execution-parameters "{\"Description\":
\"Test\",\"VpcId\":\"VPC_ID\",\"Name\":\"TestStack\",\"StackTemplateId\":
\"stm-3tdleig07sbhstgnf\",\"TimeoutInMinutes\":60,\"LoadBalancer\":{\"SecurityGroups\":
[\"sg-12345678901234567\"],\"SubnetIds\":[\"subnet-12345678901234567\"]},\"Listener1\":
{\"InstancePort\":\"80\",\"Port\":\"80\",\"Protocol\":\"HTTP\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateElbAlParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ary07xiajwx4" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateElbAlParams.json
   ```

2. Modify and save the CreateElbAlParams file. The values given in the example reflect a deployment of a Public ELB, with the health check thresholds relaxed and the ELBScheme set to `true` (for a public ELB). Note that the `Name` you set here is not the actual ELB name, you can find that name in the console as the ELB instance name. Not all optional parameters are shown in the example.

   ```
   {
     "Description" : "Test",
     "VpcId" : "VPC_ID",
     "Name" : "TestStack",
     "StackTemplateId" : "stm-3tdleig07sbhstgnf",
     "TimeoutInMinutes" : 60,
     "LoadBalancer" : {
       "SecurityGroups" : ["SG_ID, SG_ID"],
       "SubnetIds" : ["SUBNET_ID, SUBNET_ID"]
   ```

```
  },
  "Listener1" : {
    "InstancePort" : "80",
    "Port" : "80",
    "Protocol" : "HTTP"
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateElbAlRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateElbAlRfc.json
```

4. Modify and save the CreateElbAlRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0ary07xiajwx4",
"Title":                "My-ELB-Create-RFC"
}
```

5. Create the RFC, specifying the CreateElbAlRfc file and the CreateElbAlParams file:

```
aws amscm create-rfc --cli-input-json file://CreateElbAlRfc.json --execution-
parameters file://CreateElbAlParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the load balancer, look in the execution output: Use the `stack_id` to view the ELB in the Cloud Formation console or to create a Delete Stack RFC, use the ELBCName value to programmatically access the ELB.

Tips

To learn more about AWS Classic Load Balancers, see What Is a Classic Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ary07xiajwx4.

Example: Required Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-3tdleig07sbhstgnf",
  "TimeoutInMinutes" : 60,
  "LoadBalancer" : {
    "SecurityGroups" : ["sg-12345678901234567"],
    "SubnetIds" : ["subnet-12345678901234567"]
  },
  "Listener1" : {
    "InstancePort" : "80",
    "Port" : "80",
    "Protocol" : "HTTP"
  }
}
```

Example: All Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678",
  "Name" : "TestStack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-3tdleig07sbhstgnf",
  "TimeoutInMinutes" : 60,
  "LoadBalancer" : {
    "Name" : "testLoadBalancer",
    "Scheme" : "false",
    "SecurityGroups" : ["sg-12345678"],
    "SubnetIds" : ["subnet-12345678"],
    "AccessLogInterval" : "60",
    "ConnectionDrainingTimeout" : 60,
    "IdleTimeout" : 60,
    "CrossZone" : "true",
    "HealthCheckHealthyThreshold" : "2",
```

```
    "HealthCheckInterval" : "10",
    "HealthCheckTarget" : "TCP:80",
    "HealthCheckTimeout" : "5",
    "HealthCheckUnhealthyThreshold" : "10",
    "BackendInstances" : ["i-12345678"],
    "LBCookieExpirationPeriod" : "2",
    "LBCookieStickinessPolicyName" : "LBCOOKIE",
    "AppCookieName" : "APPCOOKIE",
    "AppCookiePolicyName" : "app-cookie"
  },
  "Listener1" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "80",
    "Protocol" : "HTTP",
    "PolicyNames" : ["cookie4"],
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "Listener2" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "80",
    "Protocol" : "HTTP",
    "PolicyNames" : ["cookie4", "sslPolicy"],
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "Listener3" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "80",
    "Protocol" : "HTTP",
    "PolicyNames" : ["cookie4", "sslPolicy", "two"],
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "Listener4" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "80",
    "Protocol" : "HTTP",
    "PolicyNames" : ["cookie4", "sslPolicy"],
```

```
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "Listener5" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "80",
    "Protocol" : "HTTP",
    "PolicyNames" : ["cookie4", "sslPolicy"],
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

Network Load Balancer | Create

Use to create a Network Load Balancer.

Full classification: Deployment | Advanced stack components | Network Load Balancer | Create

Change Type Details

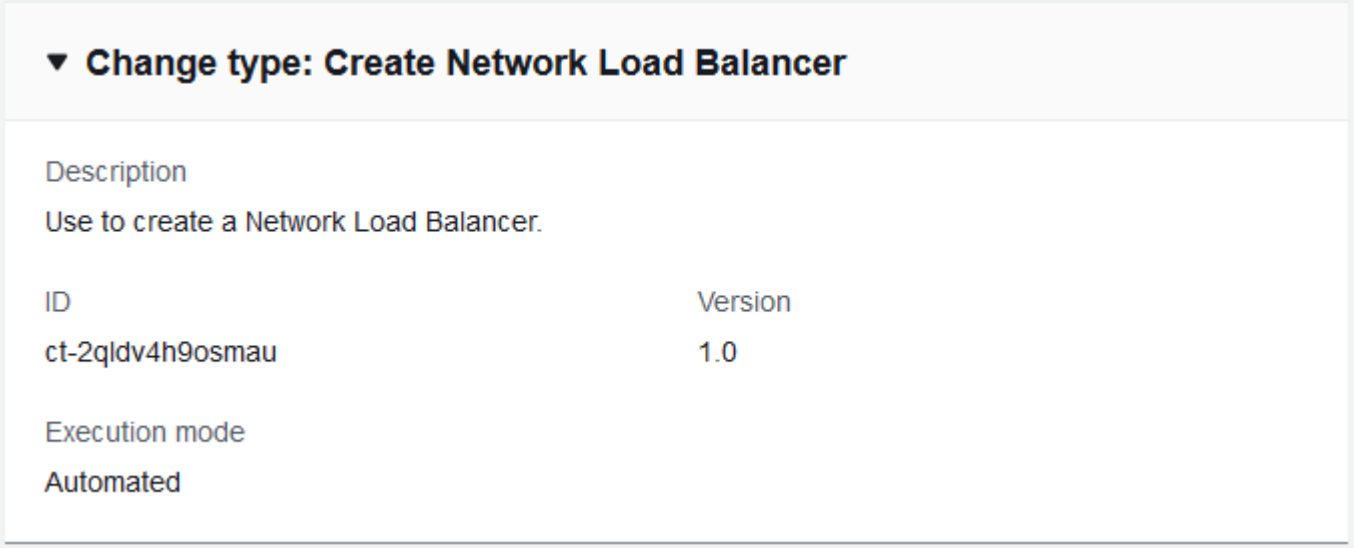
Change type ID	ct-2qldv4h9osmau
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create NLB load balancer

Creating an NLB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an NLB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc  --change-type-id
 "ct-2qldv4h9osmau" --change-type-version "1.0" --title "Test-NLB-QC" --execution-
parameters "{\"Description\":\"QCNLB\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
\"stm-l70qr9itukvqssg8d\", \"Name\":\"QCNLB\", \"TimeoutInMinutes\":60, \"Parameters\":
{\"SubnetIds\":[\"SUBNET_ID\",\"SUBNET_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateNlbParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2qldv4h9osmau" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateNlbParams.json
   ```

2. Modify and save the CreateNlbParams file. The values given in the example reflect a deployment of a public Network Load Balancer, with the health check thresholds relaxed and the `Public` parameters set to true (for a public NLB). Note that the `Name` you set here is not the actual NLB name, you can find that name in the console as the NLB instance name.

   ```
   {
   "Description":      "NLB-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-l70qr9itukvqssg8d",
   "Name":             "My-NLB",

   "Parameters":   {
       "SubnetIds":  ["PUBLIC_AZ1", "PUBLIC_AZ2"],
       "HealthCheckHealthyThreshold":   2,
       "HealthCheckInterval":           30,
       "HealthCheckTargetPath":         traffic-port",
       "DeregistrationDelayTimeout":                10,
       "Public":                        true
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it CreateNlbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateNlbRfc.json
```

4. Modify and save the CreateNlbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2qldv4h9osmau",
"Title":                "NLB-Create-RFC"
}
```

5. Create the RFC, specifying the CreateNlbRfc file and the CreateNlbParams file:

```
aws amscm create-rfc --cli-input-json file://CreateNlbRfc.json --execution-
parameters file://CreateNlbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the load balancer, look in the execution output: Use the stack_id to view the NLB in the CloudFormation console or to create a Delete Stack RFC, and use the NLB CName value to programmatically access the NLB.

Tips

> **Note**
>
> You can specify up to four Target IDs, Ports, and Availability Zones.

To learn more about AWS Network Load Balancers, see Create a Network Load Balancer.

To create a network load balancer listener, see Target Group | Create (For NLB).

To create a network load balancer target group, see Create NLB target group.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2qldv4h9osmau.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "Name": "Test Stack",
  "Parameters": {
    "CrossZoneEnabled": "false",
    "DeregistrationDelayTimeoutSeconds": "300",
    "HealthCheckHealthyThreshold": "3",
    "HealthCheckIntervalSeconds": "30",
    "HealthCheckTargetPath": "/",
    "HealthCheckTargetPort": "80",
    "HealthCheckTargetProtocol": "TCP",
    "InstancePort": "80",
    "LoadBalancerName": "my-load-balancer",
    "LoadBalancerPort": "80",
    "ProxyProtocolV2": "false",
    "Public": "false",
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"],
    "Target1AvailabilityZone": "us-east-1a",
    "Target1ID": "i-01234567890abcdef",
    "Target1Port": "80",
    "Target2AvailabilityZone":  "us-east-1a",
    "Target2ID": "i-11234567890abcdef",
    "Target2Port": "80",
    "Target3AvailabilityZone": "us-east-1a",
    "Target3ID": "i-21234567890abcdef",
    "Target3Port": "80",
    "Target4AvailabilityZone": "us-east-1a",
    "Target4ID": "i-31234567890abcdef",
    "Target4Port": "80",
    "TargetType": "instance"
  },
  "StackTemplateId": "stm-l70qr9itukvqssg8d",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef"
}
```

OpenSearch | Create Domain

Create an Amazon OpenSearch Service domain. An OpenSearch domain encapsulates OpenSearch engine instances that process OpenSearch requests. Amazon OpenSearch Service supports OpenSearch and legacy Elasticsearch OSS (up to 7.10, the final open source version of the software).

Full classification: Deployment | Advanced stack components | OpenSearch | Create domain

Change Type Details

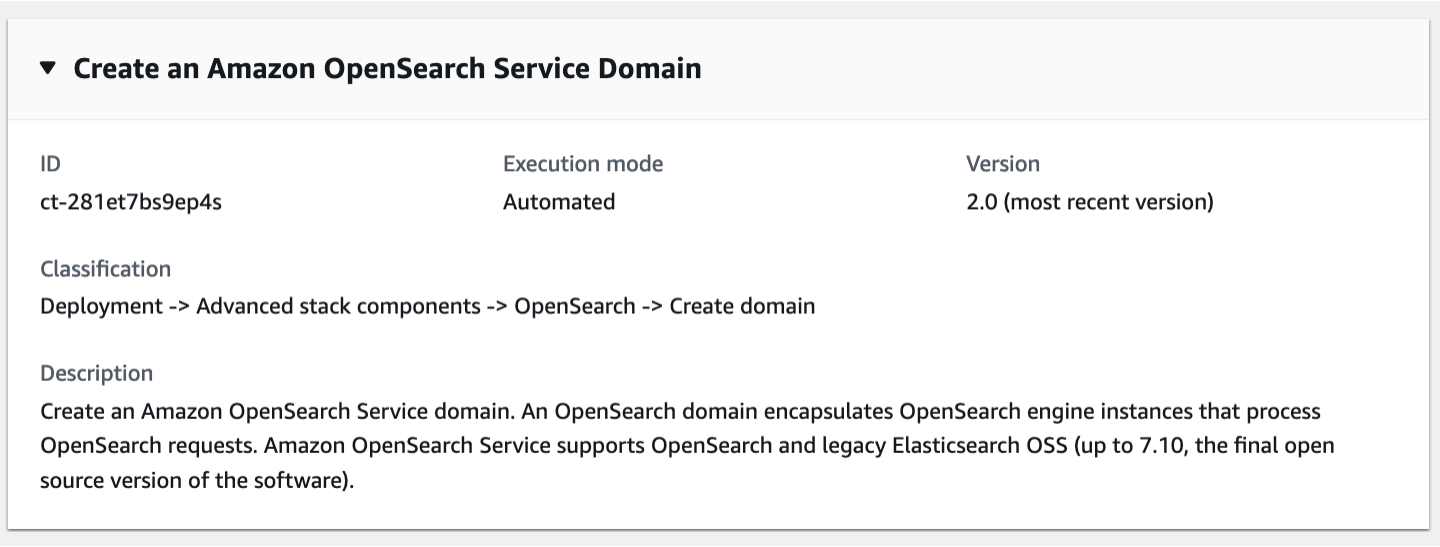
Change type ID	ct-281et7bs9ep4s
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create OpenSearch Service Domain

Creating an OpenSearch domain with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an OpenSearch domain with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-281et7bs9ep4s" \
--change-type-version "2.0" --title "Create OpenSearch domain" \
--execution-parameters "{\"Description\":\"Create OS domain\",\"VpcId\":
\"vpc-317a9856\",\"Name\": \"OpenSearchDomain\",\"StackTemplateId\": \"stm-
szccoe01000000000\",\"TimeoutInMinutes\": 60,\"Parameters\":{\"DomainName\":
\"opensearchdomain\",\"EngineVersion\":\"OpenSearch_7.10\",\"DedicatedMasterCount
\": \"3\",\"DedicatedMasterType\": \"r5.xlarge.search\",\"InstanceType\":
 \"r5.xlarge.search\",\"InstanceCount\": 2\"EBSIops\": \"0\",\"EBSVolumeSize
\": 100\"EBSVolumeType\": \"gp2\",\"SubnetIds\":[\"subnet-d3cf52f9\"]}}" \
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateOpenSearchDomainParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-281et7bs9ep4s"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateOpenSearchDomainParams.json
   ```

2. Modify and save the CreateOpenSearchDomainParams file. See examples below; make sure to modify these parameters to meet your specific needs.

   ```
   {
     "Description": "OpenSearch Service Domain",
     "VpcId": "vpc-317a9856",
     "Name": "open_search",
     "StackTemplateId": "stm-szccoe02000000000",
     "TimeoutInMinutes": 60,
     "Parameters": {
         "DomainName": "my-opensearch-domain",
         "EngineVersion": "OpenSearch_2.3",
         "DedicatedMasterCount": "3",
         "DedicatedMasterType": "r6g.large.search",
         "InstanceType": "r6g.large.search",
         "InstanceCount": "2",
           "EBSVolumeSize": "35",
           "EBSVolumeType": "gp3",
   ```

```
        "SubnetIds": [
            "subnet-0123456789abcdefg"
        ]
     }
    }
```

3. Output the RFC template to a file in your current folder; this example names it CreateOpenSearchDomainRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateOpenSearchDomainRfc.json
```

4. Modify and save the CreateOpenSearchDomainRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-281et7bs9ep4s",
    "ChangeTypeVersion": "1.0",
    "Title": "Create OpenSearch domain"
}
```

5. Create the RFC, specifying the CreateOpenSearchDomainRfc file and the CreateOpenSearchDomainParams file:

```
aws amscm create-rfc --cli-input-json file://CreateOpenSearchDomainRfc.json --
execution-parameters file://CreateOpenSearchDomainParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This change type replaces the former change type, ct-0azen3a9anxzj.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-281et7bs9ep4s.

Example: Required Parameters

```
{
```

```
  "Description": "Test description",
  "VpcId": "vpc-12345678",
  "Name": "teststak",
  "StackTemplateId": "stm-szccoe02000000000",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DomainName": "testdomain",
    "EngineVersion": "Elasticsearch_7.10",
    "DedicatedMasterCount": "0",
    "DedicatedMasterType": "r6g.large.search",
    "InstanceType": "r6g.large.search",
    "InstanceCount": 1,
    "EBSIops": "1000",
    "EBSThroughput": "125",
    "EBSVolumeSize": 100,
    "EBSVolumeType": "gp2",
    "SubnetIds": [
      "subnet-12345678",
      "subnet-13456789"
    ]
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description",
  "VpcId": "vpc-12345678",
  "Name": "teststak",
  "StackTemplateId": "stm-szccoe02000000000",
  "TimeoutInMinutes": 60,
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "Parameters": {
    "DomainName": "testdomain",
```

```
    "EngineVersion": "Elasticsearch_7.10",
    "DedicatedMasterCount": "0",
    "DedicatedMasterType": "r6g.large.search",
    "InstanceType": "r6g.large.search",
    "InstanceCount": 1,
    "ZoneAwarenessEnabled": "false",
    "EBSIops": "1000",
    "EBSThroughput": "125",
    "EBSVolumeSize": 100,
    "EBSVolumeType": "gp2",
    "EncryptionKey": "default",
    "AutomatedSnapshotStartHour": "",
    "AllowExplicitIndex": "true",
    "IndicesFieldDataCacheSize": "20",
    "MaxClauseCount": "512",
    "SubnetIds": [
      "subnet-12345678"
    ]
  }
}
```

RDS Database Stack | Create

Create an Amazon Relational Database Service (RDS) DB instance. To provision an Aurora single instance or multi-AZ instances, use CT ct-2jvzjwunghrhy.

Full classification: Deployment | Advanced stack components | RDS database stack | Create

Change Type Details

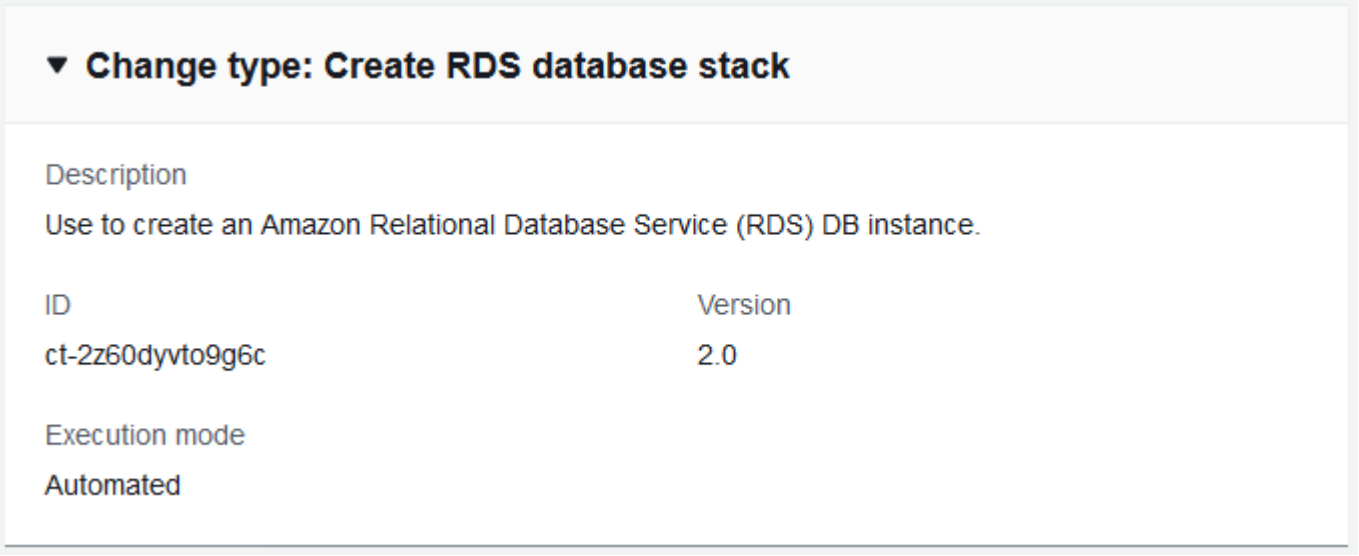
Change type ID	ct-2z60dyvto9g6c
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DB stack

Creating an RDS Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
create-rfc --change-type-id "ct-2z60dyvto9g6c" --change-type-version "3.0" --
title "RDS-Create-QC-RFC" --execution-parameters "{\"Description\":\"My RDS DB\",
```

```
 \"VpcId\":\"VPC_ID\", \"StackTemplateId\":\"stm-sl81ze00000000000\", \"Name\":
\"RDS-Create-QC\", \"TimeoutInMinutes\":60, \"Parameters\":{\"RDSAllocatedStorage
\":100, \"RDSDBEngine\":\"MySQL\", \"RDSDBName\":\"MyDB\", \"RDSEngineVersion\":
\"8.0.20\", \"RDSLicenseModel\":\"bring-your-own-license\", \"RDSMasterUsername\":
\"myUser\", \"RDSMasterUserPassword\":\"myPassWord\", \"RDSSubnetIds\":[\"SUBNET_ID\",
\"SUBNET_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2z60dyvto9g6c" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateRdsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

   ```
   {
   "Description":          "Create-RDS-DB",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-sl81ze00000000000",
   "Name":                 "My-RDS-DB",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "RDSAllocatedStorage":      50,
       "RDSDBEngine":              "oracle-se1",
       "RDSDBName":                "MyRds",
       "RDSEngineVersion":         "11.2.0.4.v13",
       "RDSInstanceType":          "db.m1.small",
       "RDSLicenseModel":          "license-included",
       "RDSMasterUsername":        "dbadmin",
       "RDSMasterUserPassword":    "p4ssw0rd",
       "RDSSubnetIds":             ["PRIVATE_AZ1_SUBNET", "PRIVATE_AZ2_SUBNET"]
       }
   }
   ```

   MySQL example:

   ```
   {
   ```

```
"Description":          "Create-RDS-DB",
"VpcId":                "VPC_ID",
"StackTemplateId":      "stm-sl81ze00000000000",
"Name":                 "My-RDS-DB",
"TimeoutInMinutes":     60,
"Parameters":   {
    "RDSAllocatedStorage":      50,
    "RDSDBEngine":              "MySQL",
    "RDSDBName":                "MyRds",
    "RDSEngineVersion":         "8.0.20",
    "RDSInstanceType":          "db.m1.small",
    "RDSLicenseModel":          "general-public-license",
    "RDSMasterUsername":        "dbadmin",
    "RDSMasterUserPassword":    "p4ssw0rd",
    "RDSSubnetIds":             ["PRIVATE_AZ1_SUBNET", "PRIVATE_AZ2_SUBNET"]
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsRfc.json
```

4. Modify and save the CreateRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-2z60dyvto9g6c",
"Title":                "RDS-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsRfc.json --execution-
parameters file://CreateRdsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console. To create a Delete Stack or Update RDS RFC, use the first part

of the DatabaseEndpoint (the DB instance ID) to create a Reboot RDS RFC, use the entire DatabaseEndpoint to programmatically access the RDS DB.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> The **RDSDBEngine** parameter has a new value available: **mariadb**.

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To create an Aurora RDS stack, see Create DB stack (for Aurora).

To create an RDS stack from a snapshot, see Create DB from snapshot.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2z60dyvto9g6c.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-sl81ze20000000000",
  "Name": "Test Stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "RDSAllocatedStorage": 50,
```

```
    "RDSDBEngine": "mysql",
    "RDSDBName": "my_db",
    "RDSEngineVersion": "5.6.27",
    "RDSInstanceType": "db.m3.medium",
    "RDSMasterUsername": "myadminuser",
    "RDSMasterUserPassword": "MySecurePassword",
    "RDSSubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-sl81ze20000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "RDSAllocatedStorage": 100,
    "RDSBackupRetentionPeriod": 7,
    "RDSAutoMinorVersionUpgrade": true,
    "RDSCharacterSetName": "",
    "RDSDBEngine": "sqlserver-ex",
    "RDSDBName": "my_db",
    "RDSDBParameterGroupName": "default.sqlserver-ex-13.0",
    "RDSDeletionProtection": true,
    "RDSEngineVersion": "13.00.4522.0.v1",
    "RDSInstanceType": "db.t2.micro",
    "RDSIOPS": 0,
    "RDSLicenseModel": "license-included",
    "RDSMasterUsername": "myadminuser",
    "RDSMasterUserPassword": "MySecurePassword",
```

```
    "RDSMultiAZ": false,
    "RDSOptionGroupName": "default:sqlserver-ex-13-00",
    "RDSPerformanceInsights" : "true",
    "RDSPerformanceInsightsKMSKey":"arn:aws:kms:us-east-1:123456789012:key/2590cd3a-
f979-49db-adec-d213775385af",
    "RDSPerformanceInsightsRetentionPeriod":"7",
    "RDSPort": 1433,
    "RDSPreferredBackupWindow": "22:00-23:00",
    "RDSPreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "RDSStorageEncrypted": true,
    "RDSStorageEncryptionKey": "arn:aws:kms:us-east-1:123456789012:key/2590cd3a-
f979-49db-adec-d213775385af",
    "RDSStorageType": "gp2",
    "RDSSubnetIds": ["subnet-12345678", "subnet-23456789"],
    "RDSTimezone": "Eastern Standard Time"
  }
}
```

RDS Database Stack | Create (For Aurora)

Create an AWS Relational Database Service (RDS) Aurora stack using either multi-availability zone (MultiAZ) or a single instance.

Full classification: Deployment | Advanced stack components | RDS database stack | Create (for Aurora)

Change Type Details

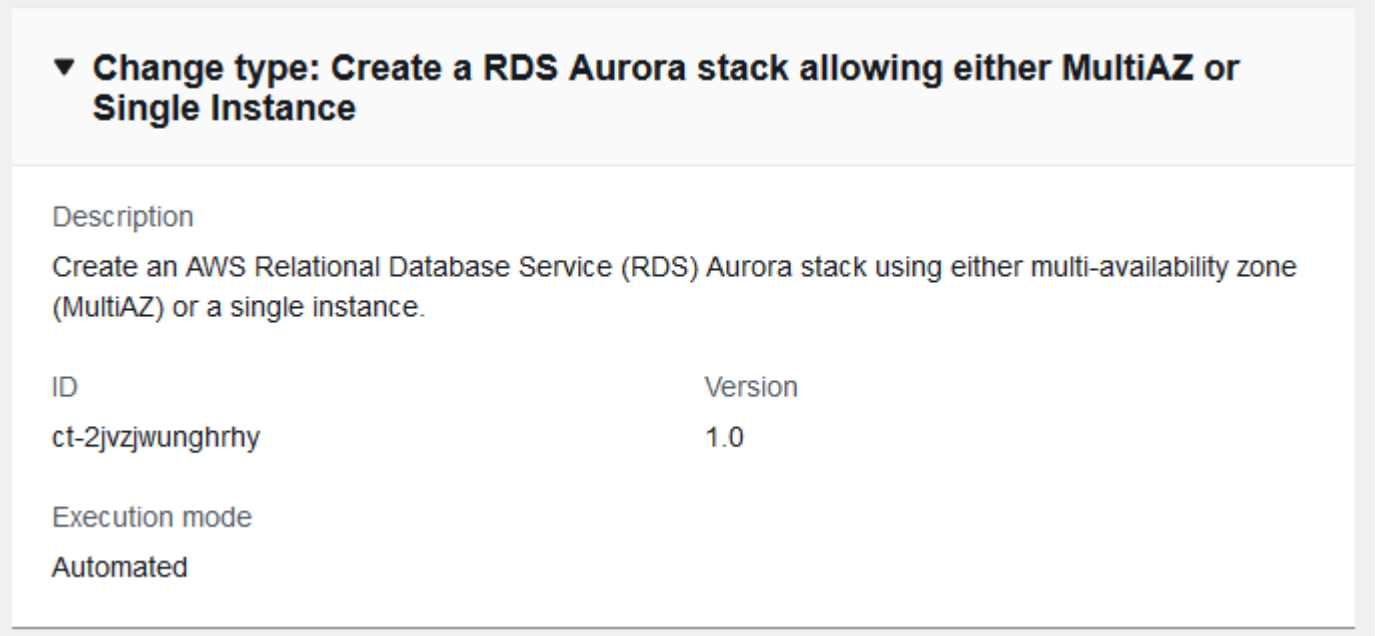
Change type ID	ct-2jvzjwunghrhy
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DB stack (for Aurora)

Creating an Aurora RDS Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Aurora RDS Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-2jvzjwunghrhy" --change-type-version "1.0" --title "Test Create (for Aurora)" --
execution-parameters "{\"Description\":\"Aurora_RDS_TEST\",\"VpcId\":\"VPC_ID\",\"Name
\":\"Aurora-TEST\",\"StackTemplateId\":\"stm-j24cifrdi0untnsn6\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"AutoMinorVersionUpgrade\":\"true\",\"BackupRetentionPeriod\":7,
\"ClusterName\":\"\",\"DBEngine\":\"aurora\",\"DBName\":\"DB_NAME\",\"DBSubnetGroupName
\":\"DB_SUBNET_GROUP_NAME\",\"EngineVersion\":\"\",\"InstanceType\":\"db.r4.large\",
\"MasterUsername\":\"DB_USER\",\"MasterUserPassword\":\"DB_PW\",\"MultiAZ\":
\"true\",\"PerformanceInsights\":\"true\",\"PerformanceInsightsKMSKey\":\"\",
\"PerformanceInsightsRetentionPeriod\":\"7\",\"Port\":\"0\",\"PreferredBackupWindow
\":\"22:00-23:00\",\"PreferredMaintenanceWindow\":\"wed:03:32-wed:04:02\",
\"StorageEncryptionKey\":\"\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsArParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2jvzjwunghrhy" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateRdsArParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

   ```
   {
     "Description": "Aurora_RDS_TEST",
     "VpcId": "VPC_ID",
     "Name": "Aurora-TEST",
     "StackTemplateId": "stm-j24cifrdi0untnsn6",
   ```

```
  "TimeoutInMinutes": 60,
  "Parameters": {
    "AutoMinorVersionUpgrade": "true",
    "BackupRetentionPeriod": 7,
    "ClusterName": "",
    "DBEngine": "aurora",
    "DBName": "DB_NAME",
    "DBSubnetGroupName": "DB_SUBNET_GROUP_NAME",
    "EngineVersion": "",
    "InstanceType": "db.r4.large",
    "MasterUsername": "DB_USER",
    "MasterUserPassword": "DB_PW",
    "MultiAZ": "true",
    "PerformanceInsights": "true",
    "PerformanceInsightsKMSKey": "",
    "PerformanceInsightsRetentionPeriod": "7",
    "Port": "0",
    "PreferredBackupWindow": "22:00-23:00",
    "PreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "StorageEncryptionKey": ""
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsArRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsArRfc.json
```

4. Modify and save the CreateRdsArRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2jvzjwunghrhy",
"Title":                "RDS-Create-Aurora-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsArRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsArRfc.json --execution-
parameters file://CreateRdsArParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console. To create a Delete Stack or Update RDS RFC, use the first part of the DatabaseEndpoint (the DB instance ID) to create a Reboot RDS RFC, use the entire DatabaseEndpoint to programmatically access the RDS DB.
7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

For more information, see Amazon Aurora – Relational Database Built for the Cloud - AWS.

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To create an Aurora RDS stack from a backup, see Create DB stack from backup (for Aurora).

To create a non-Aurora RDS stack, see Update DB stack.

To create an non-Aurora RDS stack from a snapshot, see Create DB from snapshot.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2jvzjwunghrhy.

Example: Required Parameters

```
{
  "Description": "Create an RDS stack for an Aurora database (DB) with multiple
 availability zones (MultiAZ) or as a single instance DB. Also creates an Aurora DB
 cluster consisting of a DB instance, compatible with either MySQL or PostgreSQL, and
```

```
 a cluster volume that represents the data for the DB cluster, copied across three
 Availability Zones as a single, virtual volume. The DB cluster contains a primary
 instance and, optionally, up to 15 Aurora Replicas.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-j24cifrdi0untnsn6",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DBEngine": "aurora",
    "EngineVersion": "",
    "DBName": "dbname",
    "DBSubnetGroupName": "db-subnet-group",
    "MasterUsername": "dbusername",
    "MasterUserPassword": "dbpassword"
  }
}
```

Example: All Parameters

```
{
  "Description": "Create an RDS stack for an Aurora database (DB) with multiple
 availability zones (MultiAZ) or as a single instance DB. Also creates an Aurora DB
 cluster consisting of a DB instance, compatible with either MySQL or PostgreSQL, and
 a cluster volume that represents the data for the DB cluster, copied across three
 Availability Zones as a single, virtual volume. The DB cluster contains a primary
 instance and, optionally, up to 15 Aurora Replicas.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-j24cifrdi0untnsn6",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
```

```
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "AutoMinorVersionUpgrade": "true",
    "BackupRetentionPeriod": 7,
    "ClusterName": "dbcluster",
    "DBEngine": "aurora-postgresql",
    "EngineVersion": "10.4",
    "DBName": "dbname",
    "DBSubnetGroupName": "db-subnet-group",
    "InstanceType": "db.r4.large",
    "MasterUsername": "dbusername",
    "MasterUserPassword": "dbpassword",
    "MultiAZ": "true",
    "PerformanceInsights": "true",
    "PerformanceInsightsKMSKey": "default",
    "PerformanceInsightsRetentionPeriod": "7",
    "Port": "1150",
    "PreferredBackupWindow": "22:00-23:00",
    "PreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "StorageEncryptionKey": "default"
  }
}
```

RDS Database Stack | Create DB Subnet Group

Create a Relational Database Service (RDS) database (DB) subnet group to be used with a specified RDS DB.

Full classification: Deployment | Advanced stack components | RDS database stack | Create DB subnet group

Change Type Details

Change type ID	ct-17w6f6kzf6w51

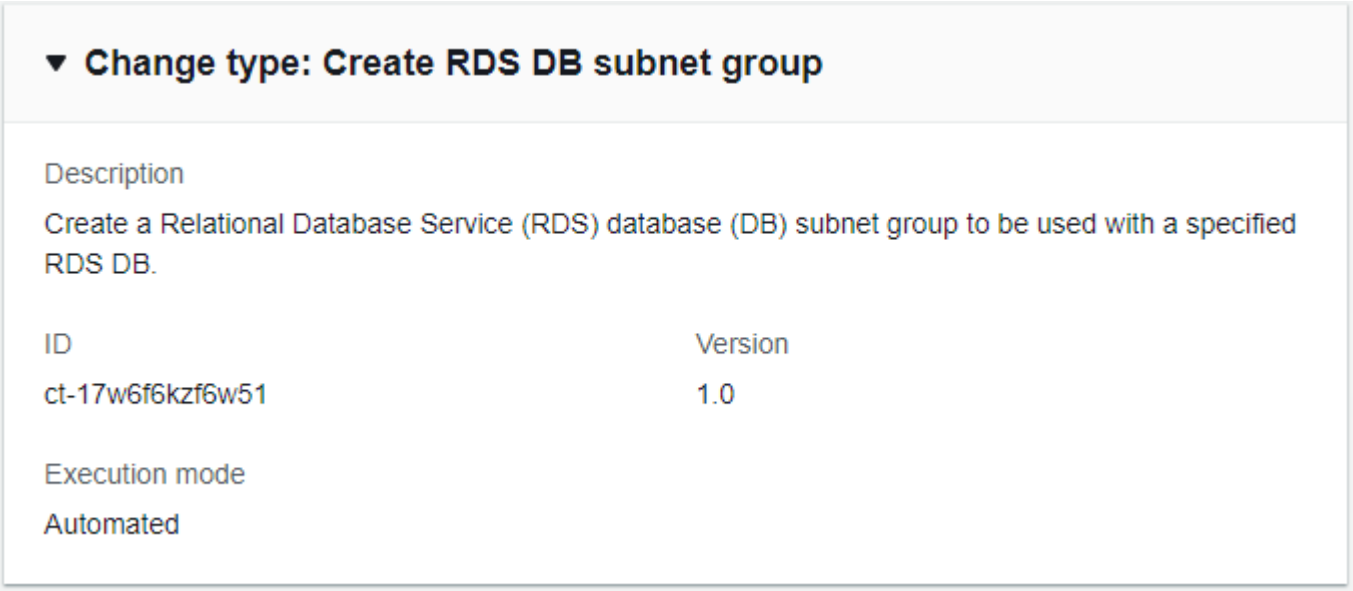
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DB subnet groups

Creating an RDS DB Subnet Group with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS DB Subnet Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
create-rfc --change-type-id "ct-17w6f6kzf6w51" --change-type-version "1.0" --title
 "RDS-subnet-group" --execution-parameters "{\"Description\":\"RDS DB subnet group\",
 \"VpcId\":\"VPC_ID\", \"StackTemplateId\":\"stm-iutsfv5ci7suupr86\", \"Name\":\"RDS
 subnet group\", \"TimeoutInMinutes\":60, \"Parameters\": {\"DBSubnetGroupName\":
 \"mydbsubnetgroup\", \"DBSubnetGroupDescription\": \"DbSubnetGroupDescription\",
 \"SubnetIds\": [\"SUBNET_ID_1\",\"SUBNET_ID_2\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2z60dyvto9g6c" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateRdsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "RDS DB subnet group",
     "VpcId": "VPC_ID",
   ```

```
  "Name": "RDS DB subnet group",
  "StackTemplateId": "stm-iutsfv5ci7suupr86",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DBSubnetGroupName": "mydbsubnetgroup",
    "DBSubnetGroupDescription": "Example RDS db subnet group description",
    "SubnetIds": [
      "SUBNET_ID_1",
      "SUBNET_ID_2"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsRfc.json
```

4. Modify and save the CreateRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-17w6f6kzf6w51",
"Title":                "RDS-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsRfc.json --execution-
parameters file://CreateRdsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Advanced** view.

To learn more about Amazon RDS DB subnet groups, see Working with an Amazon RDS DB Instance in a VPC.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-17w6f6kzf6w51.

Example: Required Parameters

```
{
  "Description": "Create RDS db subnet group",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-iutsfv5ci7suupr86",
  "Name": "Stack Name",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DBSubnetGroupName": "dbsubnetgroupname",
    "SubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
  "Description": "Create RDS db subnet group",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-iutsfv5ci7suupr86",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DBSubnetGroupName": "dbsubnetgroupname",
```

```
    "DBSubnetGroupDescription": "Test description",
    "SubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

RDS Database Stack | Create from Backup

Create an Amazon Relational Database Service (RDS) from a backup. When you restore a backup this way, the service-specific restore parameters are presented automatically.

Full classification: Deployment | Advanced stack components | RDS database stack | Create from backup

Change Type Details

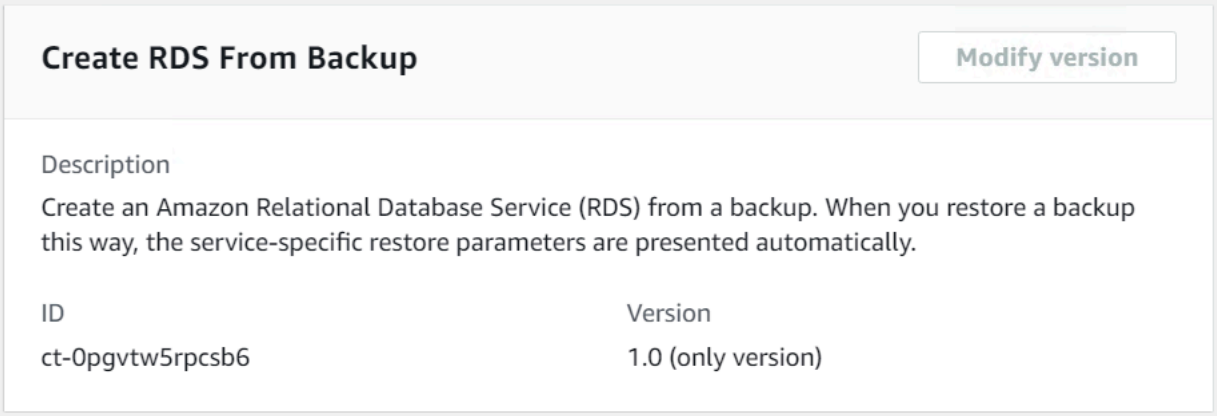
Change type ID	ct-0pgvtw5rpcsb6
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DB stack from backup

Creating an RDS stack from backup with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS stack from backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
```

```
--change-type-id "ct-0pgvtw5rpcsb6" \
--change-type-version "1.0" --title "Create RDS From Backup" \
--execution-parameters "{\"Description\":\"Create RDS Instance Stack from
 Backup: awsbackup:job-00000000-0000-0000-0000-000000000.\",\"VpcId\":
\"vpc-00000000\",\"StackTemplateId\":\"stm-siqajx00000000000\",\"Name\":
\"restoredb\",\"TimeoutInMinutes\":360,\"Parameters\":{\"DBSnapshotIdentifier
\":\"awsbackup:job-00000000-0000-0000-0000-000000000\",\"DBSubnetIds\":
[\"subnet-79663144\", \"subnet-d0cf52fa\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateRdsFromBackupParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0g690ekkyfm79"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdsFromBackupParams.json
   ```

2. Modify and save the CreateRdsFromBackupParams file.

   ```
   {
     "Description": "Create RDS Instance Stack from Backup:
    awsbackup:job-00000000-0000-0000-0000-000000000.",
     "VpcId": "vpc-00000000",
     "StackTemplateId": "stm-siqajx00000000000",
     "Name": "Stack Name",
     "TimeoutInMinutes": 360,
     "Parameters": {
       "DBSnapshotIdentifier": "awsbackup:job-00000000-0000-0000-0000-000000000",
       "DBSubnetIds": ["subnet-79663144", "subnet-d0cf52fa"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it CreateRdsFromBackupRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateRdsFromBackupRfc.json
   ```

4. Modify and save the CreateRdsFromBackupRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
    "ChangeTypeId": "ct-0pgvtw5rpcsb6",
    "ChangeTypeVersion": "1.0",
    "Title": "Create RDS Instance Stack from Backup"
}
```

5. Create the RFC, specifying the CreateRdsFromBackupRfc file and the CreateRdsFromBackupParams file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsFromBackupRfc.json --
execution-parameters file://CreateRdsFromBackupParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0pgvtw5rpcsb6.

Example: Required Parameters

```
{
  "Description": "Create RDS Instance Stack from snapshot:
 awsbackup:job-00000000-0000-0000-0000-000000000000.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-siqajx00000000000",
  "Name": "Stack Name",
  "Parameters": {
    "DBSnapshotIdentifier": "awsbackup:job-00000000-0000-0000-0000-000000000000",
    "DBSubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
```

```
  "Description": "Create RDS Instance Stack from snapshot:
 rds:sr341e8q8bofsd-2017-04-19-22-13.",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-siqajx00000000000",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "DBInstanceClass": "db.m3.medium",
    "DBInstanceIdentifier": "my-rds-id",
    "DBSnapshotIdentifier": "customizedSnapshotId",
    "DBSubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"]
  }
}
```

RDS Database Stack | Create from Backup (For Aurora)

Create an AWS Relational Database Service (RDS) Aurora stack from AWS Backup.

Full classification: Deployment | Advanced stack components | RDS database stack | Create from backup (for Aurora)

Change Type Details

Change type ID	ct-2wllq61djysxz
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required

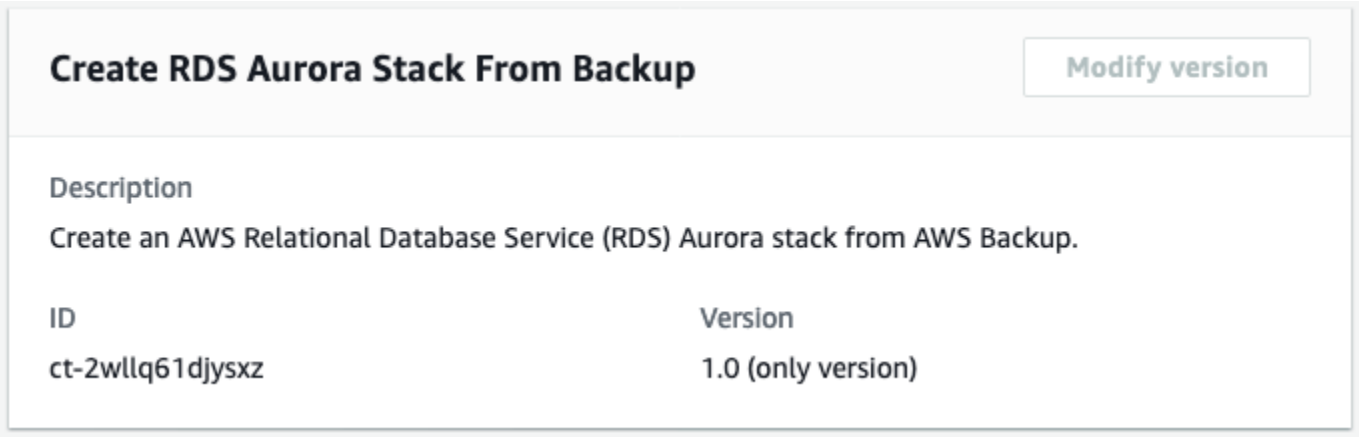
Execution mode	Automated

Additional Information

Create DB stack from backup (for Aurora)

Creating an Aurora RDS Stack From Backup with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Aurora RDS Stack From Backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-
type-id "ct-2wllq61djysxz" --change-type-version "1.0" --title
 "TestCreateAuroraStackFromBackup" --execution-parameters "{\"Description\":
\"TestCreateAuroraStackFromBackup\",\"VpcId\":\"VPC_ID\",\"Name\":\"Test Aurora
 Stack From Backup\",\"Parameters\":{\"SnapshotIdentifier\":\"SNAPSHOT_IDENTIFIER\",
\"AutoMinorVersionUpgrade\":\"true\",\"BackupRetentionPeriod\":7,\"ClusterName\":
\"\",\"DBEngine\":\"aurora\",\"DBName\":\"\",\"EngineVersion\":\"\",\"InstanceType
\":\"db.r4.large\",\"MultiAZ\":\"true\",\"Port\":\"0\",\"PreferredBackupWindow
\":\"22:00-23:00\",\"PreferredMaintenanceWindow\":\"\",\"DBSubnetGroupName
\":\"DB_SUBNET_GROUP_NAME\"},\"StackTemplateId\":\"stm-j24cifrdi0untnsn6\",
\"TimeoutInMinutes\":60}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsArFrmBkupParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2wllq61djysxz"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdsArFrmBkupParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "TestCreateAuroraStackFromBackup",
     "VpcId": "VPC_ID",
     "Name": "Test Aurora Stack From Backup",
     "Parameters": {
       "SnapshotIdentifier": "SNAPSHOT_IDENTIFIER",
       "AutoMinorVersionUpgrade": "true",
   ```

```
    "BackupRetentionPeriod": 7,
    "ClusterName": "",
    "DBEngine": "aurora",
    "DBName": "",
    "EngineVersion": "",
    "InstanceType": "db.r4.large",
    "MultiAZ": "true",
    "Port": "0",
    "PreferredBackupWindow": "22:00-23:00",
    "PreferredMaintenanceWindow": "",
    "DBSubnetGroupName": "DB_SUBNET_GROUP_NAME"
  },
  "StackTemplateId": "stm-j24cifrdi0untnsn6",
  "TimeoutInMinutes": 60
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsArFrmBkupRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsArFrmBkupRfc.json
```

4. Modify and save the CreateRdsArFrmBkupRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2wllq61djysxz",
"Title":                "RDS-Create-Aurora-From-Backup-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsArFrmBkupRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsArFrmBkupRfc.json --
execution-parameters file://CreateRdsArFrmBkupParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console. To create a Delete Stack or Update RDS RFC, use the first part of the DatabaseEndpoint (the DB instance ID); to create a Reboot RDS RFC, use the entire DatabaseEndpoint -> ClusterEndpoint to programmatically access the RDS DB.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

For more information, see Amazon Aurora – Relational Database Built for the Cloud - AWS.

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To create an Aurora RDS stack (not using backup), see Create DB stack (for Aurora).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2wllq61djysxz.

Example: Required Parameters

```
{
  "Description": "Create an AWS Relational Database Service (RDS) Aurora stack from AWS
 Backup.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-j24cifrdi0untnsn6",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
```

```
  "TimeoutInMinutes": 60,
  "Parameters": {
    "SnapshotIdentifier": "arn:aws:rds:xx-xxxx-x:000000000000:cluster-
snapshot:awsbackup:job-00000000-0000-0000-0000-000000000000",
    "DBEngine": "aurora",
    "EngineVersion": "",
    "DBSubnetGroupName": "db-subnet-group"
  }
}
```

Example: All Parameters

```
{
  "Description": "Create an AWS Relational Database Service (RDS) Aurora stack from AWS
 Backup.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-j24cifrdi0untnsn6",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "SnapshotIdentifier": "arn:aws:rds:xx-xxxx-x:000000000000:cluster-
snapshot:awsbackup:job-00000000-0000-0000-0000-000000000000",
    "AutoMinorVersionUpgrade": "true",
    "BackupRetentionPeriod": 7,
    "ClusterName": "dbcluster",
    "DBEngine": "aurora-postgresql",
    "EngineVersion": "10.4",
    "DBName": "dbname",
    "DBSubnetGroupName": "db-subnet-group",
    "InstanceType": "db.r4.large",
    "MultiAZ": "true",
    "Port": "1150",
    "PreferredBackupWindow": "22:00-23:00",
```

```
    "PreferredMaintenanceWindow": "wed:03:32-wed:04:02"
  }
}
```

RDS Database Stack | Create from Snapshot

Create an Amazon Relational Database Service (RDS) DB instance from an RDS snapshot.

Full classification: Deployment | Advanced stack components | RDS database stack | Create from snapshot

Change Type Details

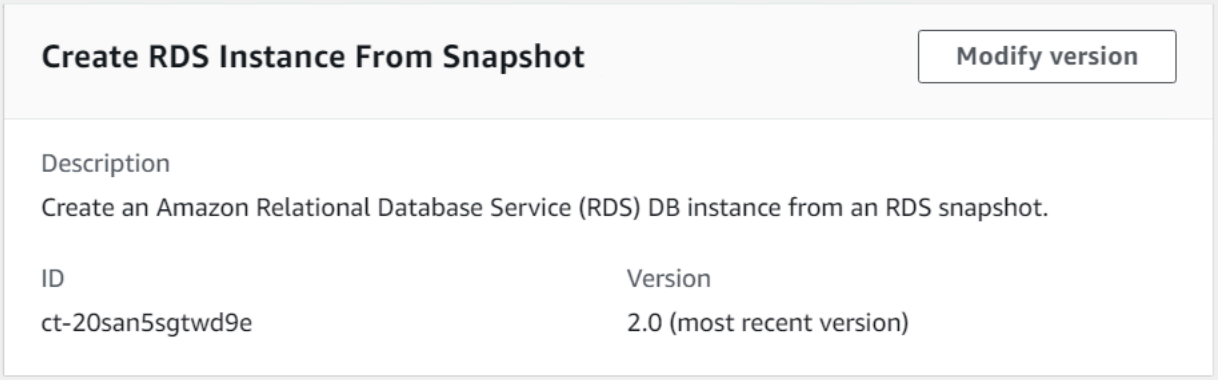
Change type ID	ct-20san5sgtwd9e
Current version	2.0
Expected execution duration	720 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DB from snapshot

Creating an RDS Stack From a Snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS Stack From a Snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Backup is enabled on RDS instances with a default backup retention period of 7 days (`RDSBackups` and `RDSBackupRetentionPeriod`).

RDS stacks do not require a grant access RFC to access them, and are instead accessed using the username and password you provided when you created the stack.

> **Note**
>
> You can add up to 50 tags.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
create-rfc --change-type-id "ct-20san5sgtwd9e" --change-type-version "2.0" --title
 "RDS-Create-FrmSS-QC-RFC" --execution-parameters "{\"Description\":\"My RDS DB
 From SS\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":\"stm-siqajx00000000000\",
 \"Name\":\"RDS-Create-FrmSS-QC\", \"TimeoutInMinutes\":60, \"Parameters\":
{ \"DBSnapshotIdentifier\":\"DB_ID\",  \"DBSubnetIds\":[\"SUBNET_ID\",\"SUBNET_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type (ct-20san5sgtwd9e) to a JSON file named CreateRdsFSParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-20san5sgtwd9e" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateRdsFSParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

   ```
   {
   "Description":          "Create-RDS-DB",
   "VpcId":                "VPC_ID",
   "StackTemplateId":      "stm-siqajx00000000000",
   "Name":                 "My-RDS-DB",
   "TimeoutInMinutes":     60,
   "Parameters":   {
       "DBSnapshotIdentifier":      rds:memzlbcde0abcd-2018-05-21-11-58,
       "DBInstanceIdentifer":      "MyRds",
       "DBSubnetIds":             ["PRIVATE_AZ1_SUBNET", "PRIVATE_AZ2_SUBNET"]
       }
   ```

```
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsRfc.json
```

4. Modify and save the CreateRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-20san5sgtwd9e",
"Title":                "RDS-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsRfc.json --execution-
parameters file://CreateRdsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console. To create a Delete Stack or Update RDS RFC, use the first part of the DatabaseEndpoint (the DB instance ID) to create a Reboot RDS RFC, use the entire DatabaseEndpoint to programmatically access the RDS DB.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> You can't restore a DB instance from a DB snapshot that is both shared and encrypted. Instead, you can make a copy of the DB snapshot and restore the DB instance from the copy. To copy the shared snapshot, use the RDS Snapshot | Copy CT.

This CT is now at version 2, with new parameters, **DBDomain**, **DBDomainIAMRoleName**, and **DBEngine**. Additionally, The v1 of the CT would initiate the stack launch and return the stackId immediately, but not wait for the stack to finish launching. The v2 of the CT waits for the stack to finish launching before marking the RFC as successful or failed.

You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more about Amazon RDS, see Amazon Relational Database Service Documentation.

To create a non-Aurora RDS stack, see RDS database stack | Create.

To create an Aurora RDS stack, see RDS database stack | Create (For Aurora).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-20san5sgtwd9e.

Example: Required Parameters

```
{
  "Description": "Create RDS Instance Stack from snapshot:
 rds:sr341e8q8bofsd-2017-04-19-22-13.",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-siqajx20000000000",
  "Name": "Stack Name",
  "TimeoutInMinutes": 360,
  "Parameters": {
    "DBSnapshotIdentifier": "rds:lr1jnp6dfxk6mha-2017-04-13-22-14",
    "DBSubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
  "Description": "Create RDS Instance Stack from snapshot:
 rds:sr341e8q8bofsd-2017-04-19-22-13.",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-siqajx20000000000",
  "Name": "Stack Name",
  "Tags": [
```

```
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 360,
  "Parameters": {
    "DBInstanceClass": "db.m3.medium",
    "DBInstanceIdentifier": "my-rds-id",
    "DBSnapshotIdentifier": "customizedSnapshotId",
    "DBSubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"],
    "DBDomain": "d-1234567890",
    "DBDomainIAMRoleName": "customer_amazon_rds_directory_service_access_role",
    "DBEngine": "sqlserver-se"
  }
}
```

RDS Snapshot | Copy

Create a KMS key encrypted copy of an Amazon Relational Database Service (Amazon RDS) DB snapshot. If you are copying a snapshot shared from another AWS account, it must be located in the same region in which the document is executed.

Full classification: Deployment | Advanced stack components | RDS snapshot | Copy

Change Type Details

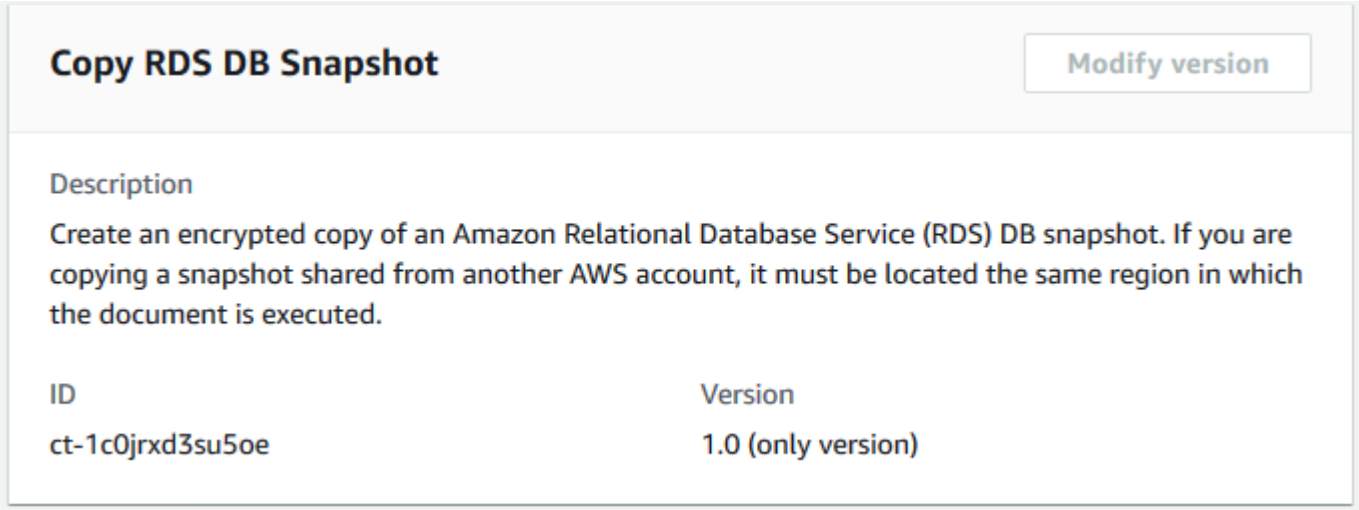
Change type ID	ct-1c0jrxd3su5oe
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Copy RDS snapshot

Copying an RDS DB Snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Copying an RDS DB Snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1c0jrxd3su5oe" --change-type-version
 "1.0" --title "Copy RDS DB snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CopyDbSnapshot\",\"Region\":\"us-east-1\",\"Parameters\":
{\"SourceDbSnapshotArn\":[\"arn:aws:rds:us-east-2:012345678901:snapshot:my-db-
snapshot\"],\"TargetDbSnapshotIdentifier\":[\"new-db-snapshot\"],\"KmsKeyId\":
[\"arn:aws:kms:us-east-1:012345678901:key/01234567-abcd-abcd-abcd-0123456789ab\"],
\"SourceRegion\":[\"us-east-2\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CopyRdsDbSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1c0jrxd3su5oe"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CopyRdsDbSnapshotParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CopyDbSnapshot",
     "Region": "us-east-1",
     "Parameters": {
       "SourceDbSnapshotArn": [
         "arn:aws:rds:us-east-2:012345678901:snapshot:my-db-snapshot"
       ],
       "TargetDbSnapshotIdentifier": [
         "new-db-snapshot"
       ],
       "KmsKeyId": [
         "arn:aws:kms:us-east-1:012345678901:key/01234567-abcd-abcd-abcd-0123456789ab"
       ],
       "SourceRegion": [
         "us-east-2"
       ]
   ```

```
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CopyRdsDbSnapshotRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CopyRdsDbSnapshotRfc.json
```

4. Modify and save the CopyRdsDbSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1c0jrxd3su5oe",
  "Title": "Copy RDS DB Snapshot"
}
```

5. Create the RFC, specifying the execution parameters file and the CopyRdsDbSnapshotRfc file:

```
aws amscm create-rfc --cli-input-json file://CopyRdsDbSnapshotRfc.json --execution-
parameters file://CopyRdsDbSnapshotParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> For more information on RDS snapshots, see Copying a DB snapshot.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1c0jrxd3su5oe.

Example: Required Parameters

```
{
```

```
  "DocumentName": "AWSManagedServices-CopyDbSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceDbSnapshotArn": ["arn:aws:rds:us-east-1:012345678901:snapshot:test-
snapshot"],
    "TargetDbSnapshotIdentifier": ["new-snapshot"],
    "KmsKeyId": ["01234567-abcd-abcd-abcd-0123456789ab"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyDbSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceDbSnapshotArn": ["arn:aws:rds:us-east-1:012345678901:snapshot:test-
snapshot"],
    "TargetDbSnapshotIdentifier": ["new-snapshot"],
    "KmsKeyId": ["01234567-abcd-abcd-abcd-0123456789ab"],
    "SourceRegion": ["us-east-1"],
    "OptionGroupName": ["my-option-group-name"]
  }
}
```

RDS Snapshot | Copy (For Aurora)

Create a copy of an Amazon Relational Database Service (Amazon RDS) DB Cluster snapshot. If you are copying a snapshot shared from another AWS account, it must be located in the same AWS Region as the specified DocumentName.

Full classification: Deployment | Advanced stack components | RDS snapshot | Copy (for Aurora)

Change Type Details

Change type ID	ct-19fdy7np55xiu
Current version	1.0
Expected execution duration	60 minutes

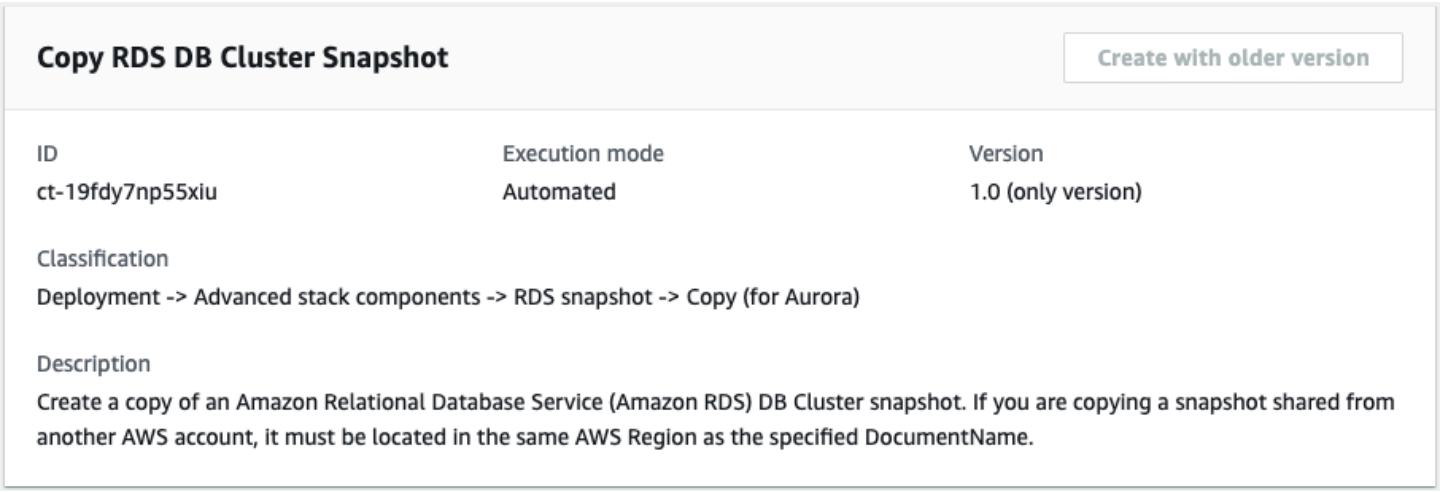
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Copy RDS Aurora snapshot

Copying an RDS Aurora snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Copying an RDS Aurora snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc  --change-type-id="ct-19fdy7np55xiu" --change-type-version="1.0"
 --title="Copy Amazon RDS Aurora snapshot" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-CopyDBClusterSnapshot\",\"Region\":\"us-east-1\",\"Parameters
\":{\"SourceDBClusterSnapshotARN\":[\"arn:aws:rds:us-east-1:111122223333:cluster-
snapshot:myauroradbcluster-snapshot\"],\"TargetDBClusterSnapshotIdentifier
\":[\"myauroradbcluster-target-snapshot\"],\"KmsKeyId\":[\"arn:aws:kms:us-
east-1:111122223333:key/01234567-abcd-abcd-abcd-0123456789ab\"],\"SourceRegion\":[\"us-
east-1\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CopyRdsAuroraSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-19fdy7np55xiu"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CopyRdsAuroraSnapshotParams.json
   ```

2. Modify and save the execution parameters in the CopyRdsAuroraSnapshotParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName":"AWSManagedServices-CopyDBClusterSnapshot",
     "Region":"us-east-1",
     "Parameters": {
       "SourceDBClusterSnapshotARN":"arn:aws:rds:us-east-1:111122223333:cluster-
   snapshot:myauroradbcluster-snapshot",
       "TargetDBClusterSnapshotIdentifier":"myauroradbcluster-target-snapshot",
   ```

```
    "KmsKeyId": "arn:aws:kms:us-east-1:111122223333:key/01234567-abcd-abcd-
abcd-0123456789ab",
    "SourceRegion":"us-east-1"
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CopyRdsAuroraSnapshotRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CopyRDSAuroraSnapshotRfc.json
```

4. Modify and save the CopyRdsAuroraSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-19fdy7np55xiu",
  "Title": "Copy Aurora snapshot"
}
```

5. Copy the RFC, specifying the input CopyDBClusterSnapshotRfc.json file and the execution parameters file CopyDBClusterSnapshotParams.json:

```
aws amscm create-rfc --cli-input-json file://CopyRdsAuroraSnapshotRfc.json --
execution-parameters file://CopyRdsAuroraSnapshotParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For an overview of backing up and restoring Aurora clusters, see Backing up and restoring an Amazon Aurora DB cluster.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-19fdy7np55xiu.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyDBClusterSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceDBClusterSnapshotARN": ["arn:aws:rds:us-east-1:111122223333:cluster-
snapshot:source-snapshot"],
    "TargetDBClusterSnapshotIdentifier": ["target-snapshot"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CopyDBClusterSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SourceDBClusterSnapshotARN": ["arn:aws:rds:us-east-1:111122223333:cluster-
snapshot:source-snapshot"],
    "TargetDBClusterSnapshotIdentifier": ["target-snapshot"],
    "KmsKeyId": ["01234567-abcd-abcd-abcd-0123456789ab"],
    "SourceRegion": ["us-east-1"]
  }
}
```

RDS Snapshot | Create

Create a snapshot of an Amazon Relational Database Service (RDS) database (DB) instance. The snapshot will be encrypted with the same KMS key as the DB instance, or unencrypted if DB instance is unencrypted.

Full classification: Deployment | Advanced stack components | RDS snapshot | Create

Change Type Details

Change type ID	ct-393q3yaq9ewlm
Current version	1.0
Expected execution duration	60 minutes

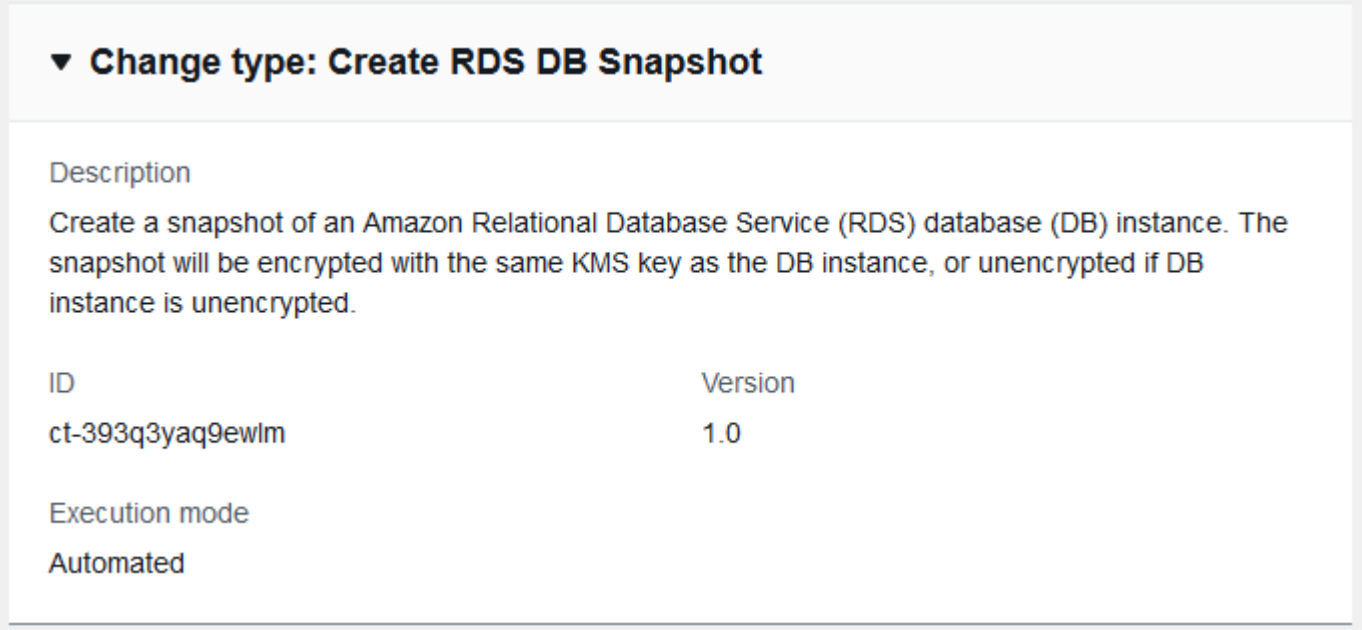
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create RDS snapshot

Creating an RDS DB Snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS DB Snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-393q3yaq9ewlm" --change-type-version
 "1.0" --title "Create DB snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CreateDBSnapshot\",\"Region\":\"us-east-1\",\"Parameters
\":{\"DBInstanceIdentifier\":[\"rds-db-instance\"],\"DBSnapshotName\":[\"my-db-
snapshot\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsDbSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-393q3yaq9ewlm"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdsDbSnapshotParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateDBSnapshot",
     "Region": "us-east-1",
     "Parameters": {
       "DBInstanceIdentifier": [
         "rds-db-instance"
   ```

```
    ],
    "DBSnapshotName": [
      "my-db-snapshot"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsDbSnapshotRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsDbSnapshotRfc.json
```

4. Modify and save the CreateRdsDbSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-393q3yaq9ewlm",
  "Title": "Create DB snapshot"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsDbSnapshotRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsDbSnapshotRfc.json --
execution-parameters file://CreateRdsDbSnapshotParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-393q3yaq9ewlm.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDBSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": ["dbinstance"],
    "DBSnapshotName": ["test-snapshot"]
  }
}
```

RDS Snapshot | Create (For Cluster)

Create a snapshot of Amazon Aurora or Multi-AZ DB (Amazon RDS) cluster in available state. The snapshot will be encrypted with the same KMS key as the DB cluster, or unencrypted if the DB cluster is unencrypted.

Full classification: Deployment | Advanced stack components | RDS snapshot | Create (for cluster)

Change Type Details

Change type ID	ct-2zqwr34epwzx1
Current version	1.0
Expected execution duration	180 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create RDS cluster snapshot

Creating an RDS DB cluster snapshot with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an RDS DB cluster snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2zqwr34epwzx1" --change-type-version
 "1.0" --title "Create a snapshot of a RDS DB Cluster" --execution-parameters
 "{\"DocumentName\": \"AWSManagedServices-CreateDBClusterSnapshot\",\"Region\":
 \"us-east-1\",\"Parameters\": {\"DBClusterIdentifier\": [\"testdbcluster\"],
\"DBClusterSnapshotIdentifier\": [\"testdbcluster-snapshot\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdsDbClusterSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2zqwr34epwzx1"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdsDbClusterSnapshotParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-CreateDBClusterSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "DBClusterIdentifier" : [ "testdbcluster" ],
    "DBClusterSnapshotIdentifier" : [ "testdbcluster-snapshot" ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it CreateRdsDbClusterSnapshotRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdsDbClusterSnapshotRfc.json
```

4. Modify and save the CreateRdsDbClusterSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2zqwr34epwzx1",
  "Title": "Create DB cluster snapshot"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdsDbClusterSnapshotRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdsDbClusterSnapshotRfc.json --
execution-parameters file://CreateRdsDbClusterSnapshotParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2zqwr34epwzx1.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDBClusterSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "DBClusterIdentifier": ["dbcluster"],
    "DBClusterSnapshotIdentifier": ["dbclustersnapshotname"]
  }
}
```

Redshift | Create (Cluster from Snapshot)

Create a Redshift cluster with the same configration as the source snapshot.

Full classification: Deployment | Advanced stack components | Redshift | Create (cluster from snapshot)

Change Type Details

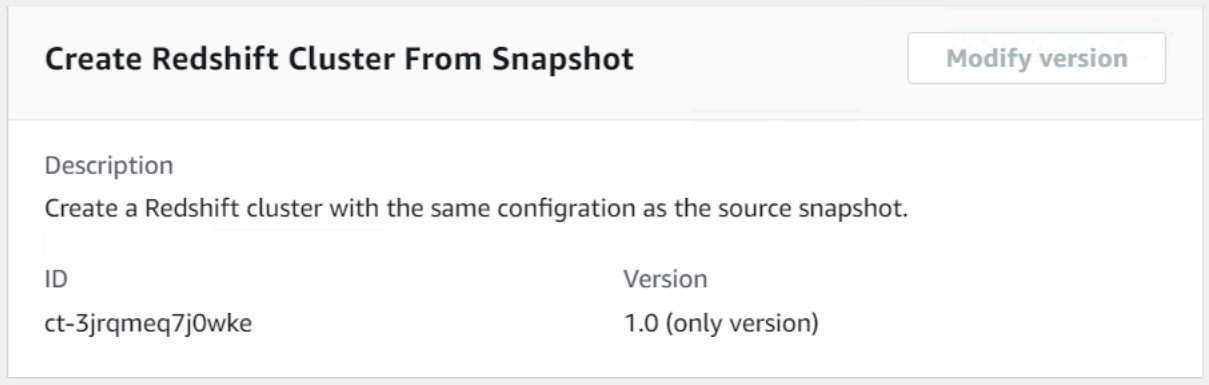
Change type ID	ct-3jrqmeq7j0wke
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create cluster from snapshot

Creating a Redshift Cluster from a snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Redshift Cluster from a snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
amscm create-rfc --change-type-id "ct-3jrqmeq7j0wke” --change-type-version
 "1.0" --title "Create redshift cluster from snapshot"  --execution-parameters
 "{\"Description\": \"DESCRIPTION\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
 \"stm-szovkq00000000000\", \"Name\":\"loremipsum\",\"TimeoutInMinutes\": 60,
 \"Parameters\": {\"AllowVersionUpgrade\": \"false\",\"DatabasePortNumber\": 5439,
\"AutomatedSnapshotRetentionPeriod\": 7,\"PreferredMaintenanceWindow\": \"\",
\"ClusterSnapshot\": \"mysnapshot\",\"ClusterSubnetGroup\": \"mysubnetgroup\"}
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateRdshftClusterFromSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3jrqmeq7j0wke"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdshftClusterFromSnapshotParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

   ```
   {
         "Description" : "Create a Redshift cluster from a snapshot",
         "VpcId" : "vpc-12345678901234567",
         "Name" : "TestStack",
         "StackTemplateId" : "stm-szovkq00000000000",
         "TimeoutInMinutes" : 60,
         "Parameters" : {
           "ClusterSnapshot" : "mysnapshot",
           "ClusterSubnetGroup" : "mysubnetgroup"
         }
       }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateRdshftClusterFromSnapshotRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton >
 CreateRdshftClusterFromSnapshotRfc.json
```

4. Modify and save the CreateRdshftClusterFromSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3jrqmeq7j0wke",
"Title":                "Redshift-Cluster-Create-From-Snapshot-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdshftClusterFromSnapshotRfc file:

```
aws amscm create-rfc --cli-input-json file://
CreateRdshftClusterFromSnapshotRfc.json --execution-parameters file://
CreateRdshftClusterFromSnapshotParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more about AWS Redshift snapshots, see Amazon Redshift snapshots.

To create an AWS Redshift cluster subnet group, see Create cluster subnet group.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3jrqmeq7j0wke.

Example: Required Parameters

```
{
  "Description" : "Create a Redshift cluster from a snapshot",
  "VpcId" : "vpc-12345678901234567",
  "Name": "Teststack",
  "StackTemplateId" : "stm-szovkq00000000000",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "ClusterSnapshot" : "mysnapshot",
    "ClusterSubnetGroup": "mysubnetgroup",
    "NodeType": "ds2.xlarge"
  }
}
```

Example: All Parameters

```
{
  "Description" : "Create a Redshift cluster from a snapshot",
  "VpcId" : "vpc-12345678901234567",
  "Name": "Teststack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-szovkq00000000000",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "ClusterIdentifier" : "test1223",
    "ClusterSnapshot" : "mysnapshot",
    "SnapshotAccountOwner" : "123456789101",
    "SnapshotClusterIdentifier" : "mycluster",
    "NodeType": "ds2.xlarge",
    "IamRoles" : "arn:aws:iam::123456789012:role/customer_redshift_role",
    "ParameterGroupName" : "myparamgroup",
    "ClusterSubnetGroup" : "mysubnetgroup",
    "AllowVersionUpgrade" : "false",
    "SecurityGroups" : ["sg-12345678"],
    "DatabasePortNumber" : 5439,
    "AutomatedSnapshotRetentionPeriod" : 7,
    "PreferredMaintenanceWindow" : "sat:00:00-sat:01:00"
```

```
  }
}
```

Redshift | Create (Cluster Subnet Group)

Use to create a Redshift cluster subnet group.

Full classification: Deployment | Advanced stack components | Redshift | Create (cluster subnet group)

Change Type Details

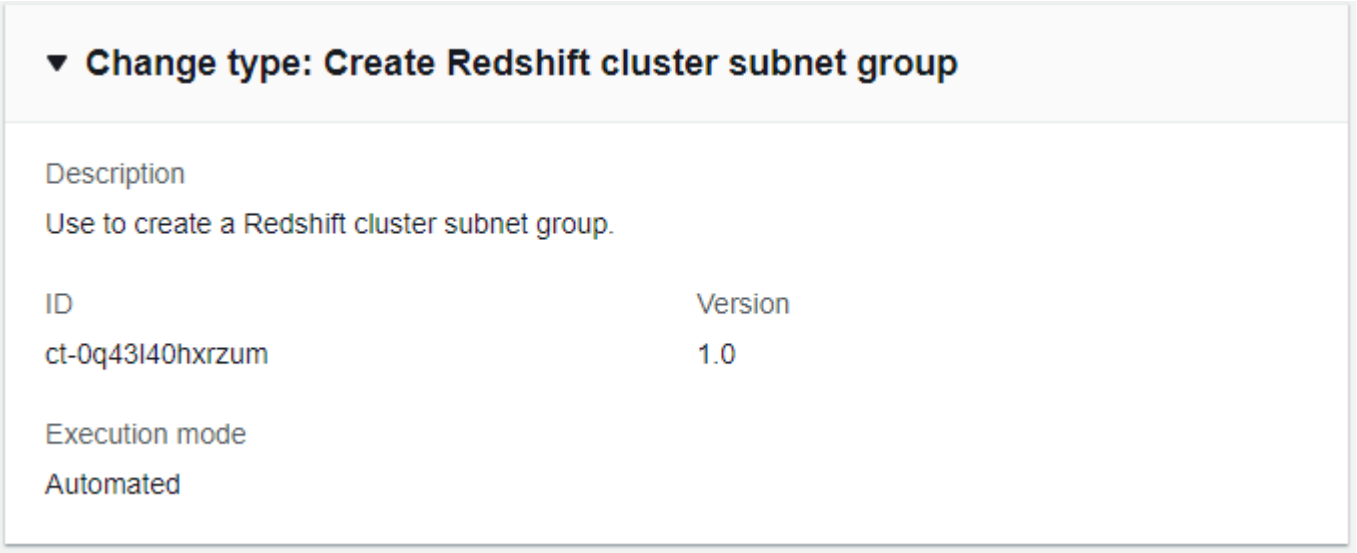
Change type ID	ct-0q43l40hxrzum
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create cluster subnet group

Creating a Redshift Cluster Subnet Group with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Redshift Cluster Subnet Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-0q43l40hxrzum" --change-type-version "1.0" --title "RedshiftClusterSubnetGroup"
 --execution-parameters "{\"Description\": \"DESCRIPTION\", \"VpcId\":\"VPC_ID\",
 \"StackTemplateId\": \"stm-5rsvv3l4760usboci\", \"Name\":\"loremipsum\",
\"TimeoutInMinutes\": 60, \"Parameters\": { \"SubnetGroupDescription\":
\"mysubnetgroup\", \"SubnetIds\": [ \"SUBNET_ID\", \"SUBNET_ID\" ] } }"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type (ct-0q43l40hxrzum) to a JSON file named CreateRedshftClusterSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0q43l40hxrzum"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRedshftClusterSGParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "Create a Redshift cluster subnet group",
     "VpcId": "VPC_ID",
     "StackTemplateId": "stm-5rsvv3l4760usboci",
     "Name": "Stack_Name",
     "TimeoutInMinutes": 60,
     "Parameters":  {
      "SubnetGroupDescription": "My Redshift Subnet Group",
      "SubnetIds": [ "SUBNET_ID", "SUBNET_ID" ]
    }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateRedshftClusterSGRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateRedshftClusterSGRfc.json
   ```

4. Modify and save the CreateRedshftClusterSGRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0q43l40hxrzum",
"Title":                "Redshift-Cluster-Create-SG-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRedshftClusterSGRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRedshftClusterSGRfc.json --
execution-parameters file://CreateRedshftClusterSGParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more about AWS Redshift, see Amazon Redshift Cluster Subnet Groups.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0q43l40hxrzum.

Example: Required Parameters

```
{
  "Description": "Create a Redshift cluster subnet group",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-5rsvv3l4760usboci",
  "Name": "Stack Name",
  "TimeoutInMinutes": 60,
```

```
  "Parameters": {
    "SubnetGroupDescription": "Test subnet group description",
    "SubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Example: All Parameters

```
{
  "Description": "Create a Redshift cluster subnet group",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-5rsvv3l4760usboci",
  "Name": "Stack Name",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "SubnetGroupDescription": "Test subnet group description",
    "SubnetIds": ["subnet-1234567890abcdef0", "subnet-1234567890abcdef1"]
  }
}
```

Redshift | Create (Cluster)

Create an Amazon Redshift cluster that is a fully managed data warehouse that consists of a set of compute nodes.

Full classification: Deployment | Advanced stack components | Redshift | Create (cluster)

Change Type Details

Change type ID	ct-1malj7snzxrkr
Current version	1.0

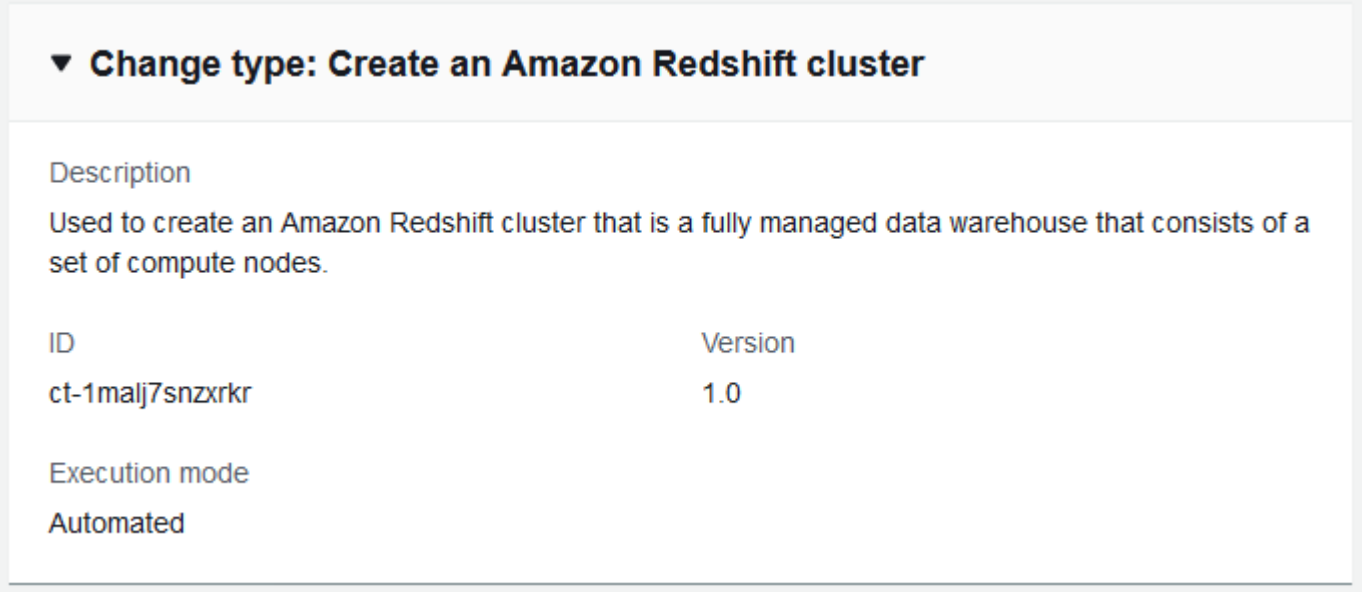
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create cluster

Creating a Redshift Cluster with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Redshift Cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1malj7snzxrkr" --change-type-version "1.0" --title "RedshiftClusterRfc"
 --execution-parameters "{\"Description\":\"DESCRIPTION\", \"VpcId\":
\"VPC_ID\", \"StackTemplateId\": \"stm-n8kpln6rtg1eiq83b\", \"Name\":
\"My_Redshift_Cluster\", \"TimeoutInMinutes\":60, \"Parameters\": {\"ClusterSubnetGroup
\":\"CLUSTER_SUBNET_GROUP\",\"DatabaseName\":\"DB_NAME\",\"MasterUsername\":\"USER\",
\"MasterUserPassword\":\"PASSWORD\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type (ct-1malj7snzxrkr) to a JSON file named CreateRdshftClusterParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1malj7snzxrkr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRdshftClusterParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

   ```
   {
     "Description": "Create a Redshift cluster",
     "VpcId": "VPC_ID",
     "StackTemplateId": "stm-n8kpln6rtg1eiq83b",
     "Name": "Stack_Name",
     "TimeoutInMinutes": 60,
     "Parameters": {
       "ClusterSubnetGroup": "mysubnetgroup",
       "DatabaseName": "myfirstdb",
       "MasterUsername": "myusername",
       "MasterUserPassword": "Mypassword1234"
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it CreateRdshftClusterRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRdshftClusterRfc.json
```

4. Modify and save the CreateRdshftClusterRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1malj7snzxrkr",
"Title":                "Redshift-Cluster-Create-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the CreateRdshftClusterRfc file:

```
aws amscm create-rfc --cli-input-json file://CreateRdshftClusterRfc.json --
execution-parameters file://CreateRdshftClusterParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more about AWS Redshift, see Amazon Redshift.

To create an AWS Redshift cluster subnet group, see Create cluster subnet group.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1malj7snzxrkr.

Example: Required Parameters

```
{
  "Description": "Create a Redshift cluster",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-n8kpln6rtg1eiq83b",
  "Name": "Stack Name",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ClusterSubnetGroup": "mysubnetgroup",
    "DatabaseName": "myfirstdb",
    "MasterUsername": "loremipsum",
    "MasterUserPassword": "Mypassword1234"
  }
}
```

Example: All Parameters

```
{
  "Description": "Create a Redshift cluster",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-n8kpln6rtg1eiq83b",
  "Name": "Stack Name",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ClusterIdentifier": "mycluster",
    "DatabaseName": "myfirstdb",
    "DatabasePortNumber": 14231,
    "MasterUsername": "loremipsum",
    "MasterUserPassword": "Mypassword1234",
    "NodeType": "ds2.xlarge",
    "ClusterType": "multi-node",
    "NumberOfNodes": "5",
    "ParameterGroupName": "myparamgroupname",
    "ClusterSubnetGroup": "mysubnetgroup",
    "SecurityGroups": ["sg-1a2b3c4d","sg-1a2b3c4d5e6f7g8h9"],
    "AllowVersionUpgrade": "true",
    "AutomatedSnapshotRetentionPeriod": 30,
    "PreferredMaintenanceWindow": "sat:00:00-sat:01:00",
    "IamRoles": "arn:aws:iam::123456789012:role/customer_redshift_role",
    "KmsKeyId": "arn:aws:kms:us-
east-1:123456789012:key/503f4b06-3507-452a-9812-7772ddc72af7"
  }
```

```
}
```

S3 Storage | Create

Create an Amazon S3 bucket for cloud storage.

Full classification: Deployment | Advanced stack components | S3 storage | Create

Change Type Details

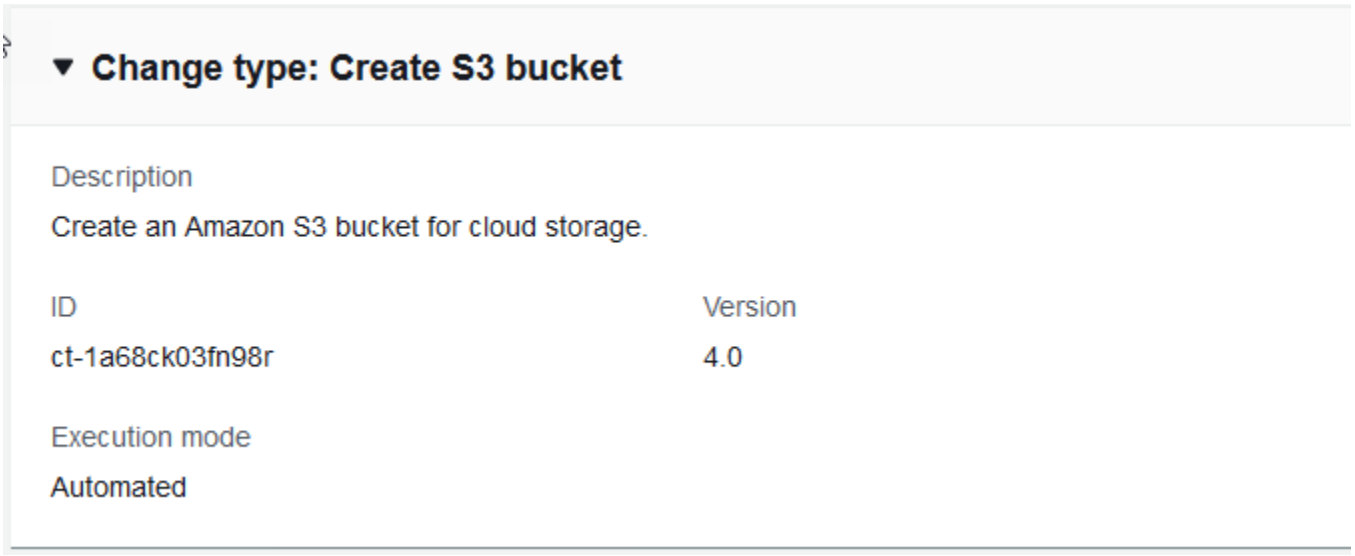
Change type ID	ct-1a68ck03fn98r
Current version	4.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create S3 storage

Creating an S3 Bucket with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an S3 Bucket with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Example with only required parameters, version 4.0 (AccessControl parameter is replaced with specific access parameters):

```
aws amscm create-rfc --title "my-s3-bucket" --change-type-id "ct-1a68ck03fn98r" --
change-type-version "4.0" --execution-parameters "{\"Description\":\"S3 bucket for
 application A.\",\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-s2b72beb200000000\",
\"Name\":\"my-s3-bucket\",\"TimeoutInMinutes\":60,\"Parameters\":{\"BucketName\":\"my-
s3-bucket\"}}"
```

Example with version 3.0 parameters including AccessControl parameter:

```
aws --profile saml amscm create-rfc --change-type-id "ct-1a68ck03fn98r" --
change-type-version "2.0" --title "TITLE" --execution-parameters "{\"Description
\":\"YOUR_S3_DESCRIPTION\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":\"stm-
s2b72beb000000000\", \"Name\":\"AMS-STACK-NAME\", \"TimeoutInMinutes\":60,\"Parameters
\": {\"AccessControl\":\"PublicRead\", \"BucketName\":\"UNIQUE-S3-BUCKET-NAME\"}}"
```

Example with all parameters, version 4.0 (AccessControl parameter is replaced with specific access parameters):

```
aws amscm create-rfc --title "my-s3-bucket" --change-type-id "ct-1a68ck03fn98r"
 --change-type-version "4.0" --execution-parameters "{\"Description\":\"S3
 bucket for application A\",\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-
s2b72beb200000000\",\"Name\":\"my-s3-bucket\",\"TimeoutInMinutes\":60,\"Parameters
\":{\"BucketName\":\"my-s3-bucket\",\"ServerSideEncryption\":\"KmsManagedKeys\",
\"KMSKeyId\":\"arn:aws:kms:us-east-1:123456789012:key/fc69dcab-d3c7-4d36-
b204-9da582ae760b\",\"Versioning\":\"Enabled\",\"IAMPrincipalsRequiringReadObjectAccess
\":[\"arn:aws:iam::123456789012:role/roleA\"],\"IAMPrincipalsRequiringWriteObjectAccess
\":[\"arn:aws:iam::123456789012:role/roleA\"],\"ServicesRequiringReadObjectAccess\":
[\"logs.us-east-1.amazonaws.com\"],\"ServicesRequiringWriteObjectAccess\":[\"logs.us-
east-1.amazonaws.com\"],\"EnforceSecureTransport\":true,\"AccessAllowedIpRanges\":
[\"1.2.3.0/24\",\"2.3.4.0/24\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateBucketParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1a68ck03fn98r" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateBucketParams.json
   ```

2. Modify and save the CreateBucketParams file. Note that you do not need to use your account ID in the `BucketName`, but it can make finding the bucket easier (remember that bucket names must be unique in the account across all regions and cannot have uppercase letters). If using this to create a tier-and-tie WordPress site, you may want to indicate that purpose when setting the `BucketName`.

   Example with version 3.0 parameters including `AccessControl`:

   ```
   {
   "Description":      "S3-Bucket-Create",
   "VpcId":            "VPC-ID",
   ```

```
"StackTemplateId":  "stm-s2b72beb000000000",
"Name":             "My-S3-Bucket",
"TimeoutInMinutes": 60,
"Parameters":   {
    "AccessControl":    "Private",
    "BucketName":       "ACCOUNT_ID.BUCKET_NAME"
    }
}
```

Example with version 4.0 new access parameters (example grants READ access to the objects in the bucket for an IAM user or a role):

```
{
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "IAMPrincipalsWithReadObjectAccess": [
        "arn:aws:iam::123456789123:role/roleA",
        "arn:aws:iam::987654321987:role/roleB"
      ]
    }
}
```

For the resulting policy, see Grants READ access for an IAM User or a Role.

Example with version 4.0 new access parameters (example grants WRITE access to the objects in the bucket for an IAM user or a role):

```
{
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "IAMPrincipalsRequiringWriteObjectAccess": [
        "arn:aws:iam::123456789123:role/roleA",
```

```
        "arn:aws:iam::987654321987:role/roleB"
      ]
    }
}
```

For the resulting policy, see Grants WRITE access for an IAM User or a Role.

Example with version 4.0 new access parameters (example grants READ access to the objects in the bucket for an AWS service):

```
{
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "ServicesRequiringWriteObjectAccess": [
        "rds.amazonaws.com",
        "logs.ap-southeast-2.amazonaws.com",
        "ec2.amazonaws.com"
      ]
    }
}
```

For the resulting policy, see Grants READ access for an AWS Service.

Example with version 4.0 new access parameters (example grants WRITE access to the objects in the bucket for an AWS service):

```
{
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "ServicesRequiringWriteObjectAccess": [
        "rds.amazonaws.com",
        "logs.ap-southeast-2.amazonaws.com",
```

```
        "ec2.amazonaws.com"
      ]
    }
}
```

For the resulting policy, see Grants WRITE access for an AWS Service.

Example with version 4.0, (enforce secure transport):

```
{
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "EnforceSecureTransport": "true"
    }
}
```

For the resulting policy, see Uses EnforceSecureTransport.

Example with version 4.0, limits access to the bucket from a set of IP ranges use <>:

```
  {
    "Description": "S3-Bucket-Create",
    "VpcId": "VPC-ID",
    "StackTemplateId": "stm-s2b72beb200000000",
    "Name": "My-S3-Bucket",
    "TimeoutInMinutes": 60,
    "Parameters": {
      "BucketName": "ACCOUNT_ID.BUCKET_NAME",
      "AccessAllowedIpRanges": [
        "1.2.3.0/24",
        "2.3.4.0/24"
      ]
    }
  }
```

For the resulting policy, see Limits Access to IP Range.

3. Output the RFC template JSON file to a file named CreateBucketRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateBucketRfc.json
```

4. Modify and save the CreateBucketRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "4.0",
"ChangeTypeId":         "ct-1a68ck03fn98r",
"Title":                "S3-Bucket-Create-RFC",
"RequestedStartTime":   "2016-12-05T14:20:00Z",
"RequestedEndTime":     "2016-12-05T16:20:00Z"
}
```

5. Create the RFC, specifying the CreateBucketRfc file and the CreateBucketParams file:

```
aws amscm create-rfc --cli-input-json file://CreateBucketRfc.json  --execution-
parameters file://CreateBucketParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the S3 bucket or load objects to it, look in the execution output: Use the `stack_id` to view the bucket in the Cloud Formation Console, use the S3BucketName to view the bucket in the S3 Console.

> **Note**
>
> When uploading objects from a non-owner account, it is mandatory to specify the `bucket-owner-full-control` ACL, that grants the bucket owner account full control over all the objects in the bucket. Example:
>
> ```
> aws s3api put-object --acl bucket-owner-full-control --
> bucket ACCOUNT_ID.BUCKET_NAME --key data.txt --body /tmp/data.txt
> ```

Tips

> **Note**
>
> This walkthrough describes, and provides example commands for, creating an Amazon S3 storage bucket using version 4.0 of the change type (ct-1a68ck03fn98r). This version does not allow you to create a public S3 bucket, only private is allowed. To create a public S3 storage bucket, use a previous version of the change type, and specify **PublicRead** for the **AccessControl** parameter.
> Also, this walkthrough does not grant the permissions necessary for deleting versioned objects.

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

S3 Storage Bucket Create Resulting Policies

Depending on how you created your Amazon S3 storage bucket, you created policies. These example policies match various Amazon S3 create scenarios provided in Creating an S3 Bucket with the CLI.

Grants READ access for an IAM User or a Role

Resulting example policy grants READ access to the objects in the bucket for an IAM user or a role:

```
{
          "Sid": "AllowBucketReadActionsForArns",
          "Effect": "Allow",
          "Principal": {
              "AWS": [
                  "arn:aws:iam::123456789123:role/roleA”,
                  "arn:aws:iam::987654321987:role/roleB”
              ]
          },
          "Action": [
              "s3:GetBucketAcl",
              "s3:GetBucketLocation",
              "s3:ListBucket"
          ],
          "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME"
      },
      {
```

```
            "Sid": "AllowObjectReadActionsForArns",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::123456789123:role/roleA”,
                    "arn:aws:iam::987654321987:role/roleB”
                ]
            },
            "Action": [
                "s3:GetObject",
                "s3:ListMultipartUploadParts"
            ],
            "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*"
}
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Grants WRITE access for an IAM User or a Role

The following resulting example policy grants WRITE access to the objects in the bucket for a IAM user or a role. This policy does not grant the permissions necessary for deleting versioned objects.

```
{
            "Sid": "AllowObjectWriteActionsForArns",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::123456789123:role/roleA”,
                    "arn:aws:iam::987654321987:role/roleB”
                ]
            },
            "Action": [
                "s3:PutObject",
                "s3:DeleteObject",
                "s3:AbortMultipartUpload"
            ],
            "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*"
}
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Grants READ access for an AWS Service

Resulting example policy grants READ access to the objects in the bucket for an AWS service:

```
{
          "Sid": "AllowBucketReadActionsForServcices",
          "Effect": "Allow",
          "Principal": {
              "Service": [
                  "rds.amazonaws.com",
                  "logs.ap-southeast-2.amazonaws.com",
                  "ec2.amazonaws.com"
              ]
          },
          "Action": [
              "s3:GetBucketAcl",
              "s3:GetBucketLocation",
              "s3:ListBucket"
          ],
          "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*"
      },
      {
          "Sid": "AllowObjectReadActionsForServcices",
          "Effect": "Allow",
          "Principal": {
              "Service": [
                  "rds.amazonaws.com",
                  "logs.ap-southeast-2.amazonaws.com",
                  "ec2.amazonaws.com"
              ]
          },
          "Action": [
              "s3:GetObject",
              "s3:ListMultipartUploadParts"
          ],
          "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*"
}
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Grants WRITE access for an AWS Service

The following resulting example policy grants WRITE access to the objects in the bucket for an AWS service. This policy does not grant the permissions necessary for deleting versioned objects.

```
{
      "Sid": "AllowObjectWriteActionsForServcices",
      "Effect": "Allow",
      "Principal": {
        "Service": [
            "rds.amazonaws.com",
            "logs.ap-southeast-2.amazonaws.com",
            "ec2.amazonaws.com"
        ]
    },
    "Action": [
          "s3:PutObject",
          "s3:DeleteObject",
          "s3:AbortMultipartUpload"
      ],
      "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*"
}
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Uses EnforceSecureTransport

Resulting example policy enforcing secure transport:

```
{
          "Sid": "EnforceSecureTransport",
          "Effect": "Deny",
          "Principal": "*",
          "Action": "*",
          "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*",
          "Condition": {
              "Bool": {
                  "aws:SecureTransport": "false"
              }
          }
}
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Limits Access to IP Range

Resulting example policy limiting access to the bucket from a set of IP ranges:

```
{
          "Sid": "RestrictBasedOnIPRanges",
          "Effect": "Deny",
          "Principal": "*",
          "Action": "s3:*",
          "Resource": "arn:aws:s3:::ACCOUNT-ID.BUCKET_NAME/*",
          "Condition": {
              "NotIpAddress": {
                  "aws:SourceIp": [
                      “1.2.3.0/24",
                      “2.3.4.0/24"
                  ]
              }
          }
      }
```

For the execution parameters to create this policy with the S3 storage bucket Create change type, see Creating an S3 Bucket with the CLI

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1a68ck03fn98r.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678901234567",
  "StackTemplateId": "stm-s2b72beb200000000",
  "Name": "Test Stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "BucketName": "mybucket"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-s2b72beb200000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "BucketName": "mybucket",
    "ServerSideEncryption": "KmsManagedKeys",
    "KMSKeyId": "arn:aws:kms:ap-southeast-2:123456789012:key/9d5948f1-2082-4c07-a183-
eb829b8d81c4",
    "Versioning": "Enabled",
    "IAMPrincipalsRequiringReadObjectAccess": [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "IAMPrincipalsRequiringWriteObjectAccess": [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "ServicesRequiringReadObjectAccess": [
      "rds.amazonaws.com",
      "ec2.amazonaws.com",
      "logs.ap-southeast-2.amazonaws.com"
    ],
    "ServicesRequiringWriteObjectAccess": [
      "rds.amazonaws.com",
      "ec2.amazonaws.com",
      "logs.ap-southeast-2.amazonaws.com"
    ],
    "EnforceSecureTransport": true,
    "AccessAllowedIpRanges": [
```

```
      "1.0.0.0/24",
      "2.0.0.0/24"
    ]
  }
}
```

S3 Storage | Create Policy (Review Required)

Create an S3 bucket policy. The existing bucket policy (if any) is replaced with the new policy.

Full classification: Deployment | Advanced stack components | S3 storage | Create policy (review required)

Change Type Details

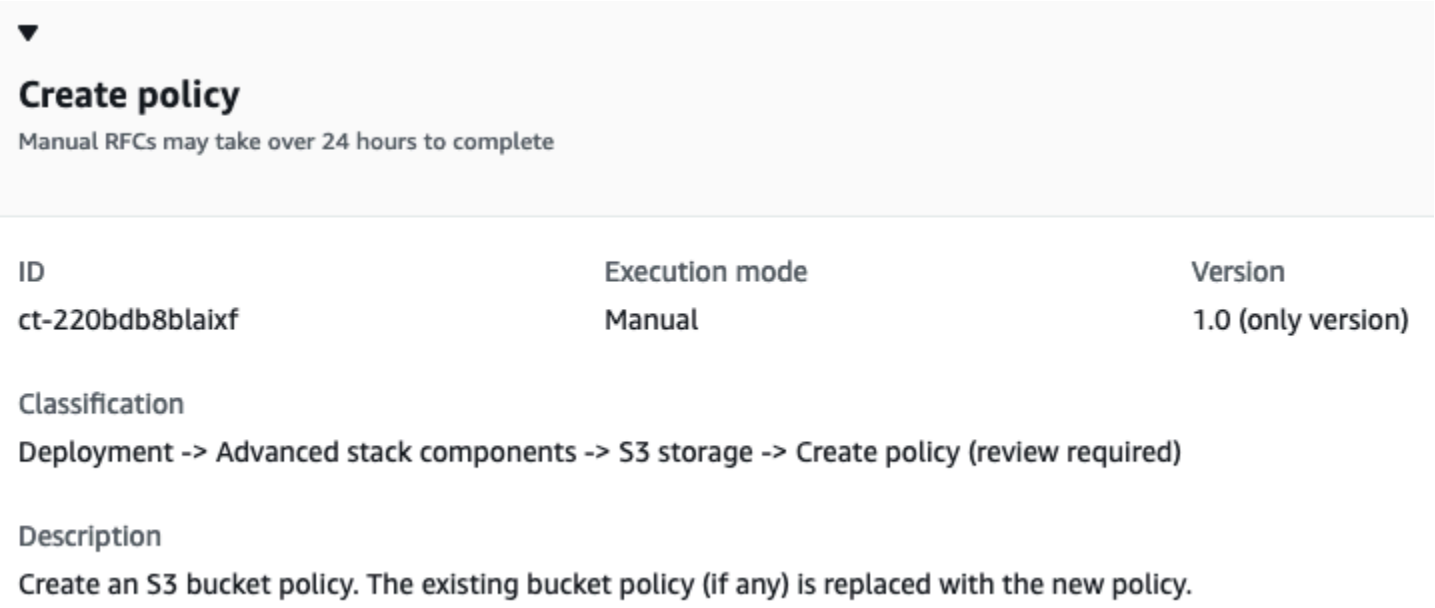
Change type ID	ct-220bdb8blaixf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create S3 storage policy

Creating an S3 Storage Policy with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an S3 Storage Policy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-220bdb8blaixf" --change-type-version "1.0"
 --title "TITLE" --execution-parameters "{\"BucketName\": \"example-bucket-123\",
 \"BucketPolicy\": \"Example bucket policy\", \"Operation\": \"Create policy\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateS3PolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-220bdb8blaixf" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateS3PolicyParams.json
   ```

2. Modify and save the CreateS3PolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "BucketName": "test-bucket-01",
     "BucketPolicy": "Bucket Policy example",
     "Operation": "Create policy"
   }
   ```

3. Output the RFC template JSON file to a file named CreateS3PolicyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateS3PolicyRfc.json
   ```

4. Modify and save the CreateS3PolicyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-220bdb8blaixf",
   "Title":                "S3-Policy-Create-RFC"
   }
   ```

5. Create the RFC, specifying the CreateS3PolicyRfc file and the CreateS3PolicyParams file:

```
aws amscm create-rfc --cli-input-json file://CreateS3PolicyRfc.json  --execution-
parameters file://CreateS3PolicyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon S3, see Using Bucket Policies and User Policies.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-220bdb8blaixf.

Example: Required Parameters

```
{
  "BucketName": "examplebucketname",
  "BucketPolicy": "Example bucket permissions",
  "Operation": "Create policy"
}
```

Example: All Parameters

```
{
  "BucketName": "examplebucketname",
  "BucketPolicy": "Example bucket permissions",
  "Operation": "Create policy",
  "Priority": "Medium"
}
```

Security Group | Create

Create a security group with limited scope. For complex security groups, use the manual Security group Create change type (ct-1oxx2g2d7hc90).

Full classification: Deployment | Advanced stack components | Security group | Create

Change Type Details

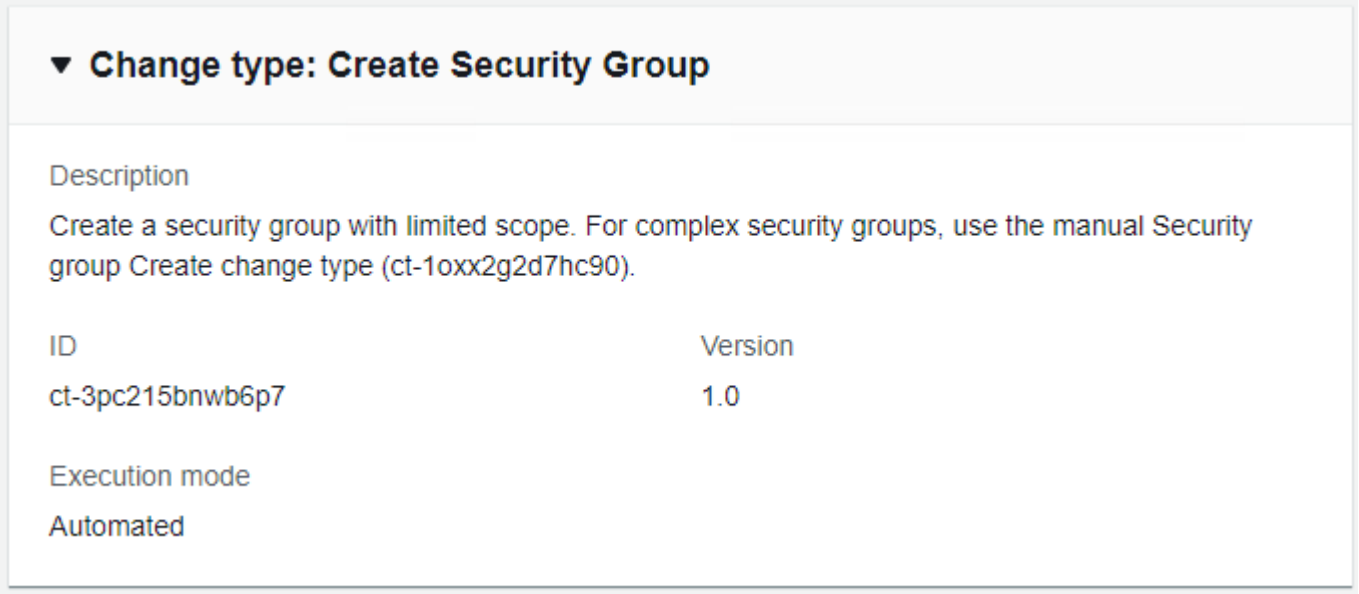
Change type ID	ct-3pc215bnwb6p7
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create security group

Creating a Security Group with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Security Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3pc215bnwb6p7" --change-
type-version "1.0" --title "Test-SG-Auto" --execution-parameters "{\"VpcId\":
\"VPC_ID\", \"Description\":\"Test-SG-Auto\", \"SecurityGroupName\":\"Test-SG-Auto\",
 \"TcpUdpIngressRules\":{\"Protocol\":\"TCP\", \"FromPort\":\"PORT\, \"ToPort\":
\"PORT\"}, \"TcpUdpEgressRules\":{\"Protocol\":\"TCP\", \"FromPort\":\"PORT\, \"ToPort
\":\"PORT\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateSGAParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3pc215bnwb6p7" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateSGAParams.json
   ```

2. Modify and save the CreateSGAParams file. For example, you can replace the contents with something like this:

   ```
   {
   "SecurityGroupName":              "My-WEB-SG",
   "SecurityGroupDescription":       "SG_DESCRIPTION",
   "TcpUdpIngressRules":   {
       "Protocol":       "PROTOCOL",
       "FromPortRange": "PORT_RANGE",
       "ToPort":         "TRAFFIC_SOURCE"
       },
   "TcpUdpEgressRules":   {
       "Protocol":      "PROTOCOL",
       "FromPort":      "PORT",
       "ToPort":        "PORT"
       }
   }
   ```

3. Output the RFC template JSON file to a file named CreateSGARfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateSGARfc.json
   ```

4. Modify and save the CreateSGARfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3pc215bnwb6p7",
"Title":                "SG-Create-Auto-RFC"
}
```

5. Create the RFC, specifying the CreateSGARfc file and the CreateSGAParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSGARfc.json  --execution-
parameters file://CreateSGAParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. Once the security group is created, use Associate security group to resource to associate the security group with your AMS resources.

Tips

> **Note**
>
> Once the security group is created, use Associate security group to resource to associate the security group with your AMS resources. In order to delete a security group, it must *not* have associated resources.

To learn more about AWS security groups and creating security groups, see Security Group Rules Reference; this page can help you determine the rules you want and, importantly, how to name your security group so choosing it when creating other resources is intuitive. Also see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

Once the security group is created, use Associate security group to resource to associate the security group with your AMS resources. In order to delete a security group, it must *not* have associated resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3pc215bnwb6p7.

Example: Required Parameters

```
{
  "VpcId": "vpc-12345678",
  "SecurityGroupName": "app1-webserver",
  "SecurityGroupDescription": "App1 group"
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-01234567890abcdef",
  "SecurityGroupName": "app1-webserver",
  "SecurityGroupDescription": "App1 group",
  "TcpUdpIngressRules": [
    { "Protocol": "TCP", "FromPort": 80, "ToPort": 80, "AddressRanges":
 ["192.168.0.0/16"], "Description": "Client1" },
    { "Protocol": "UDP", "FromPort": 80, "ToPort": 80, "SecurityGroupIds": ["sg-
abd45678901234567"], "Description": "Client1" },
    { "Protocol": "TCP", "FromPort": 80, "ToPort": 80, "AddressRanges":
 ["192.168.0.0/16"], "SecurityGroupIds": ["sg-abc45678"], "Description": "Client1" }
  ],
  "TcpUdpEgressRules": [
    { "Protocol": "TCP", "FromPort": 100, "ToPort": 120, "AddressRanges":
 ["192.168.0.0/16"], "Description": "Client1" },
    { "Protocol": "UDP", "FromPort": 100, "ToPort": 120, "SecurityGroupIds": ["sg-
abd45678901234567"], "Description": "Client1" },
    { "Protocol": "TCP", "FromPort": 100, "ToPort": 120, "AddressRanges":
 ["192.168.0.0/16"], "SecurityGroupIds": ["sg-abc45678"], "Description": "Client1" }
  ],
  "IcmpIngressRules": [
    { "Type": -1, "Code": -1, "AddressRanges": ["192.168.0.0/16"], "Description":
 "Client1" },
    { "Type": 15, "Code": 8, "SecurityGroupIds": ["sg-abd45678901234567"],
 "Description": "Client1" }
  ],
  "IcmpEgressRules": [
```

```
    { "Type": -1, "Code": -1, "AddressRanges": ["192.168.0.0/16"], "Description":
 "Client1" },
    { "Type": 30, "Code": 15, "SecurityGroupIds": ["sg-abd45678901234567"],
 "Description": "Client1" }
  ],
  "Tags": [
    { "Key": "B", "Value": "bb" },
    { "Key": "C", "Value": "cc" },
    { "Key": "D", "Value": "dd" },
    { "Key": "E", "Value": "ee" }
  ]
}
```

Security Group | Create (Review Required)

Create a security group, and optionally associate it with AWS resources.

Full classification: Deployment | Advanced stack components | Security group | Create (review required)

Change Type Details

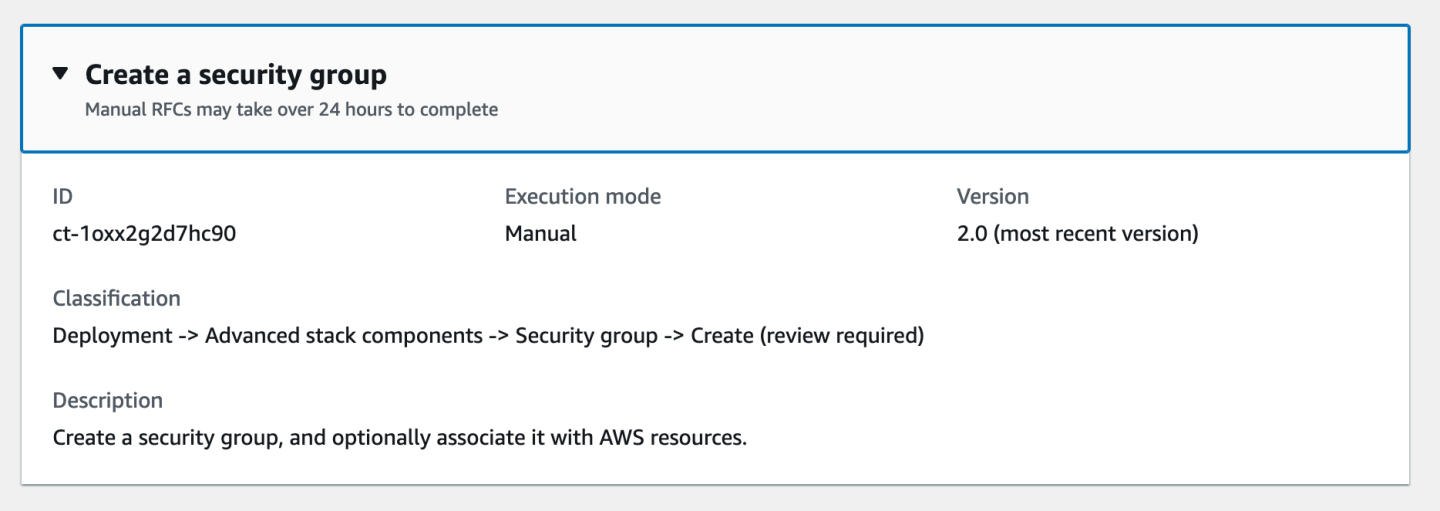
Change type ID	ct-1oxx2g2d7hc90
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create security group (review required)

Creating a Security Group (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Security Group (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-1oxx2g2d7hc90" --change-type-
version "2.0" --title "Test-SG-RR" --execution-parameters "{\"Description\":\"Test-
SG-RR\", \"Name\":\"Test-SG-IC\", \"InboundRules\":{\"Protocol\":\"TCP\", \"PortRange
\":\"49152-65535\, \"Source\":\"203.0.113.5/32\"}, \"OutboundRules\":{\"Protocol\":
\"TCP\", \"PortRange\":\"49152-65535\, \"Destination\":\"203.0.113.5/32\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateSgRrParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1oxx2g2d7hc90" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateSgRrParams.json
   ```

2. Modify and save the CreateSgRrParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "SG-Create-With-Review",
   "Name":             "My-SG",
   "VpcId":            "vpc-12345abc",
   "InboundRules":   {
       "Protocol":     "TRAFFIC_PROTOCOL",
       "PortRange":    "PORT_RANGE",
       "Source":       "TRAFFIC_SOURCE"
       },
   "OutboundRules":   {
       "Protocol":     "TRAFFIC_PROTOCOL",
       "PortRange":    "PORT_RANGE",
       "Destination":  "TRAFFIC_DESTINATION"
       }
   }
   ```

3. Output the RFC template JSON file to a file named CreateSgRrRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateSgRrRfc.json
   ```

4. Modify and save the CreateSgRrRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-1oxx2g2d7hc90",
"Title":                "SG-Create-RR-RFC"
}
```

5. Create the RFC, specifying the CreateSgRrRfc file and the CreateSgRrParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSgRrRfc.json  --execution-
parameters file://CreateSgRrParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Note

There is an automated change type for creating a security group, Deployment | Advanced stack components | Security group | Create (no review required) (ct-3pc215bnwb6p7) that provides options for TCP and ICMP ingress and egress rules. If those rules are adequate, the Create (auto) change type will execute more quickly than this change type. For details, see Security Group | Create.

Note

Once the security group is created, use Security Group | Associate to associate the security group with your AMS resources. In order to delete a security group, it must have associated resources.

Note

Outbound rules are not required; however, if they are not specified, then a "127.0.0.1/32 Blackhole Rule" is used, meaning that the resource will only be able to communicate to itself and not with other resources. You can see this default outbound rule when using the AMS console, but not when using the AMS API/CLI.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

To learn more about AWS security groups and creating security groups, see Security Group Rules Reference; this page can help you determine the rules you want and, importantly, how to name your security group so choosing it when creating other resources is intuitive. Also see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

To better understand general AWS security, see Best Practices for Security, Identity, & Compliance.

Once the security group is created, use Security Group | Associate to associate the security group with your AMS resources. In order to delete a security group, it must have associated resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1oxx2g2d7hc90.

Example: Required Parameters

```
{
  "VpcId": "vpc-12345abc",
  "Name": "app1-webserver",
  "Description": "App1 group",
  "InboundRules": [],
  "OutboundRules": []
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234abcd",
  "Name": "app1-webserver",
  "Description": "App1 group",
  "AssociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-34abcd12",
    "i-4abcd123",
```

```
   "i-abcd1234",
   "i-1234567890abcdefg",
   "i-234567890abcdefg1",
   "i-34567890abcdefg12",
   "i-4567890abcdefg123",
   "i-567890abcdefg1234"
 ],
 "InboundRules": [
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
   { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description": "Client1" },
```

```
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
```

```
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" },
    { "Protocol": "TCP", "PortRange":"80", "Source": "192.168.0.0/16", "Description":
 "Client1" }
  ],
  "OutboundRules": [
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
```

```
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
```

```
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" }
 ],
 "Priority": "Medium",
 "Tags": [
   { "Key":
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
"Value":
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
   { "Key": "B", "Value": "bb" },
   { "Key": "C", "Value": "cc" },
   { "Key": "D", "Value": "dd" },
```

```
    { "Key": "E", "Value": "ee" },
    { "Key": "F", "Value": "ff" },
    { "Key": "G", "Value": "gg" },
    { "Key": "H", "Value": "hh" },
    { "Key": "I", "Value": "ii" },
    { "Key": "J", "Value": "jj" },
    { "Key": "K", "Value": "kk" },
    { "Key": "L", "Value": "ll" },
    { "Key": "M", "Value": "mm" },
    { "Key": "N", "Value": "nn" },
    { "Key": "O", "Value": "oo" },
    { "Key": "P", "Value": "pp" },
    { "Key": "Q", "Value": "qq" },
    { "Key": "R", "Value": "rr" },
    { "Key": "S", "Value": "ss" },
    { "Key": "T", "Value": "tt" },
    { "Key": "U", "Value": "uu" },
    { "Key": "V", "Value": "vv" },
    { "Key": "W", "Value": "ww" },
    { "Key": "X", "Value": "xx" },
    { "Key": "Y", "Value": "yy" },
    { "Key": "Z", "Value": "zz" },
    { "Key": "a", "Value": "aa" },
    { "Key": "b", "Value": "bb" },
    { "Key": "c", "Value": "cc" },
    { "Key": "d", "Value": "dd" },
    { "Key": "e", "Value": "ee" },
    { "Key": "f", "Value": "ff" },
    { "Key": "g", "Value": "gg" },
    { "Key": "h", "Value": "hh" },
    { "Key": "i", "Value": "ii" },
    { "Key": "j", "Value": "jj" },
    { "Key": "k", "Value": "kk" },
    { "Key": "l", "Value": "ll" },
    { "Key": "m", "Value": "mm" },
    { "Key": "n", "Value": "nn" },
    { "Key": "o", "Value": "oo" },
    { "Key": "p", "Value": "pp" },
    { "Key": "q", "Value": "qq" },
    { "Key": "r", "Value": "rr" },
    { "Key": "s", "Value": "ss" },
    { "Key": "t", "Value": "tt" },
    { "Key": "u", "Value": "uu" },
    { "Key": "v", "Value": "vv" },
```

```
    { "Key": "w", "Value": "ww" },
    { "Key": "x", "Value": "xx" }
  ]
}
```

Storage Gateway | Create from Backup

Start an AWS Backup service restore job to restore a Storage Gateway volume snapshot of the specified resource.

Full classification: Deployment | Advanced stack components | Storage Gateway | Create from Backup

Change Type Details

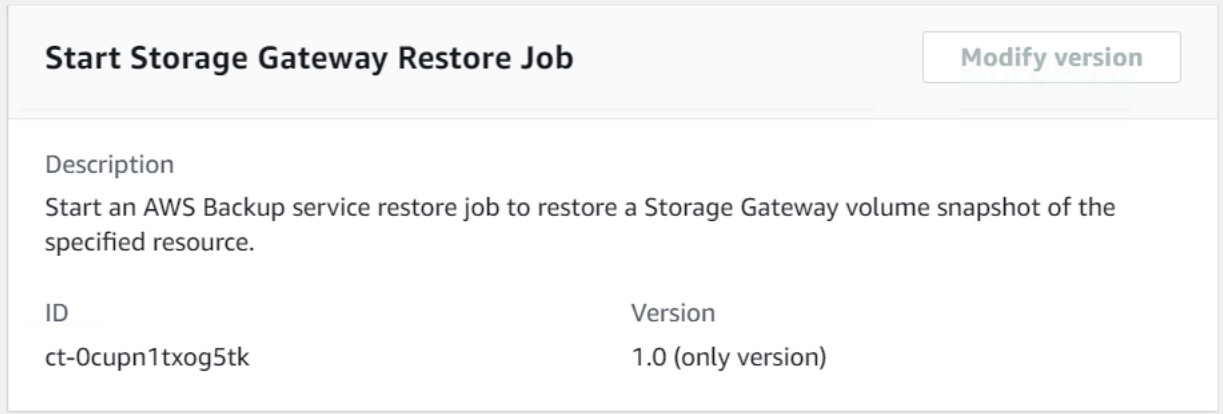
Change type ID	ct-0cupn1txog5tk
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Storage gateway, create from backup

Creating a Storage Gateway from backup with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Storage Gateway from backup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Restoring store volumes (DiskId parameter is mandatory):

```
aws amscm create-rfc \
--change-type-id "ct-0cupn1txog5tk" \
--change-type-version "1.0" --title "StartRestoreJobStorageGatewayVolume for Cached
 Volume" \
"{\"DocumentName\":\"AWSManagedServices-StartRestoreJobStorageGatewayVolume\",
\"Region\":\"us-east-1\",\"Parameters\":{\"RecoveryPointArn\":[\"arn:aws:ec2:us-
east-1::snapshot/snap-0000000000000\"],\"BackupVaultName\":[\"my-vault-name\"],
\"GatewayArn\":[\"arn:aws:storagegateway:us-east-1:000000000000:gateway/sgw-0000000\"],
\"TargetName\":[\"myTarget\"],\"GatewayType\":[\"Cached\"]}}"
```

Restoring cached volumes:

```
aws amscm create-rfc \
--change-type-id "ct-0cupn1txog5tk" \
--change-type-version "1.0" --title "StartRestoreJobStorageGatewayVolume for Stored
 Volume" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
StartRestoreJobStorageGatewayVolume\",\"Region\":\"us-east-1\",\"Parameters\":
{\"RecoveryPointArn\":[\"arn:aws:ec2:us-east-1::snapshot/snap-0000000000000\"],
\"BackupVaultName\":[\"my-vault-name\"],\"GatewayArn\":[\"arn:aws:storagegateway:us-
east-1:000000000000:gateway/sgw-0000000\"],\"TargetName\":[\"myTarget\"],\"GatewayType
\":[\"Stored\"],\"DiskId\":[\"pci-0000:00:1c.0-nvme-1\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateStoreGatewayFromBackupParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1h1tuxn2oxrtf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateStoreGatewayFromBackupParams.json
   ```

2. Modify and save the CreateStoreGatewayFromBackupParams file.

   For restore stored volumes:

   ```
   {
   "DocumentName": "AWSManagedServices-StartRestoreJobStorageGatewayVolume",
   "Region": "us-east-1",
   "Parameters": {
     "RecoveryPointArn": [
       "arn:aws:ec2:us-east-1::snapshot/snap-00000000000000"
   ```

```
  ],
  "BackupVaultName": [
    "my-vault-name"
  ],
  "GatewayArn": [
    "arn:aws:storagegateway:us-east-1:000000000000:gateway/sgw-0000000"
  ],
  "TargetName": [
    "myTarget"
  ],
  "GatewayType": [
    "Stored"
  ],
  "DiskId": [
    "pci-0000:00:1c.0-nvme-1"
  ]
}
}
```

For restore cached volumes:

```
{
"DocumentName": "AWSManagedServices-StartRestoreJobStorageGatewayVolume",
"Region": "us-east-1",
"Parameters": {
  "RecoveryPointArn": [
    "arn:aws:ec2:us-east-1::snapshot/snap-000000000000"
  ],
  "BackupVaultName": [
    "my-vault-name"
  ],
  "GatewayArn": [
    "arn:aws:storagegateway:us-east-1:000000000000:gateway/sgw-0000000"
  ],
  "TargetName": [
    "myTarget"
  ],
  "GatewayType": [
    "Cached"
  ]
}
```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateStoreGatewayFromBackupRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateStoreGatewayFromBackupRfc.json
```

4. Modify and save the CreateStoreGatewayFromBackupRfc.json file. For example, you can replace the contents with something like this:

   For restore stored volumes:

```
{
  "ChangeTypeId": "ct-0cupn1txog5tk",
  "ChangeTypeVersion": "1.0",
  "Title": "Testing ct-0cupn1txog5tk StartRestoreJobStorageGatewayVolume in region
 us-east-1 for stored volumes"
}
```

   For restore cached volumes:

```
{
  "ChangeTypeId": "ct-0cupn1txog5tk",
  "ChangeTypeVersion": "1.0",
  "Title": "Testing ct-0cupn1txog5tk StartRestoreJobStorageGatewayVolume in region
 us-east-1 for cached volumes"
}
```

5. Create the RFC, specifying the CreateStoreGatewayFromBackupRfc file and the CreateStoreGatewayFromBackupParams file:

```
aws amscm create-rfc --cli-input-json file://CreateStoreGatewayFromBackupRfc.json
 --execution-parameters file://CreateStoreGatewayFromBackupParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0cupn1txog5tk.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobStorageGatewayVolume",
  "Region": "us-east-1",
  "Parameters": {
    "RecoveryPointArn": [
      "arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"
    ],
    "BackupVaultName": [
      "Vault01"
    ],
    "GatewayArn": [
      "arn:aws:storagegateway:us-east-1:000000000000:gateway/sgw-0000000"
    ],
    "TargetName": [
      "mytarget"
    ],
    "GatewayType": [
      "Cached"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartRestoreJobStorageGatewayVolume",
  "Region": "us-east-1",
  "Parameters": {
    "RecoveryPointArn": [
      "arn:aws:ec2:us-east-1::snapshot/snap-00000000000000000"
    ],
    "BackupVaultName": [
      "Vault01"
    ],
```

```
    "GatewayArn": [
      "arn:aws:storagegateway:us-east-1:000000000000:gateway/sgw-0000000"
    ],
    "TargetName": [
      "mytarget"
    ],
    "GatewayType": [
      "Cached"
    ],
    "DiskId": [
      "pci-0000:00:1c.0"
    ],
    "VolumeSize": [
      "0"
    ],
    "IamRoleArn": [
      "arn:aws:iam::123456789012:role/my_role"
    ],
    "KmsKeyArn": [
      "arn:aws:kms:us-east-1:000000000000:key/00000000-0000-0000-0000-000000000000"
    ]
  }
}
```

Tag | Create

Add tags to existing, supported resources: Autoscaling, EC2, Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally, CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS infrastructure stacks (stacks named mc-*) cannot have tags added with this change type.

Full classification: Deployment | Advanced stack components | Tag | Create

Change Type Details

Change type ID	ct-3cx7we852p3af
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required

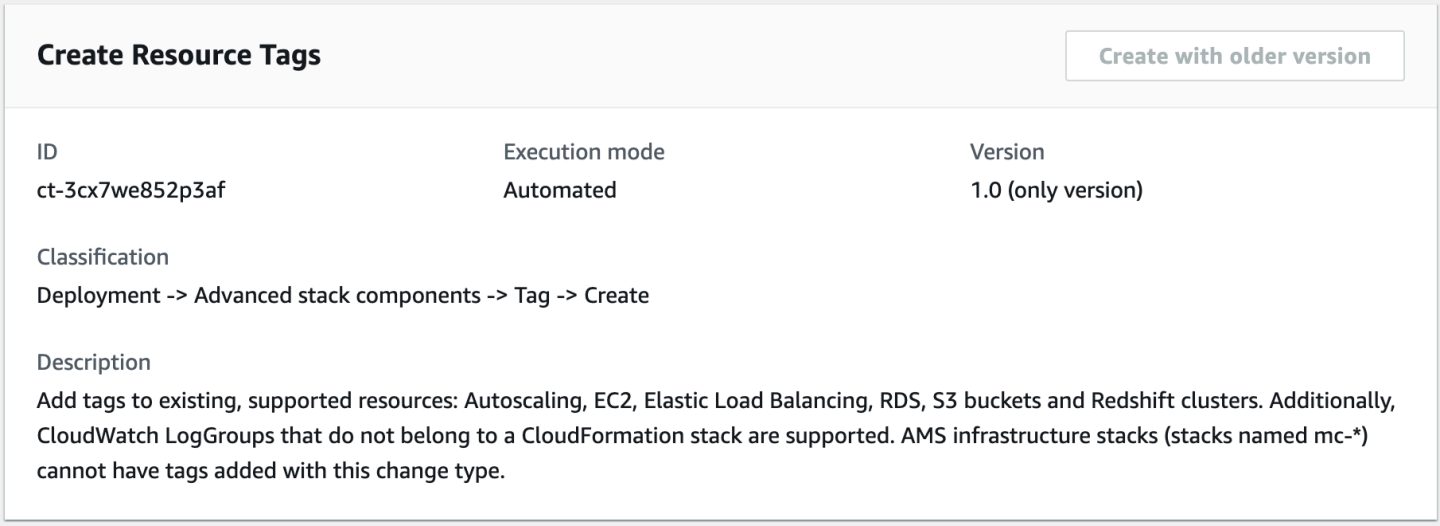
Customer approval	Not required
Execution mode	Automated

Additional Information

Create tags

Creating tags with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating tags with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3cx7we852p3af" --change-type-version
 "1.0" --title "Create Tags" --execution-parameters --execution-parameters
 '{"DocumentName":"AWSManagedServices-UpdateTags","Region":"us-east-1","Parameters":
{"ResourceArns":["i-1234567890abcdef0","vol-1234567890abcdef0","arn:aws:rds:us-
east-1:123456789012:db/my-db-instance"],"AddOrUpdateTags":["{\"Key\":\"Name\",\"Value
\":\"App1\"}","{\"Key\":\"Owner\",\"Value\":\"Dev\"}"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagCreateAutoParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagCreateAutoParams.json
   ```

2. Modify and save the TagCreateAutoParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateTags",
     "Region": "us-east-1",
     "Parameters":     {
       "ResourceArns": [
         "i-1234567890abcdef0",
         "vol-1234567890abcdef0",
         "arn:aws:rds:us-east-1:123456789012:db/my-db-instance"
       ],
       "AddOrUpdateTags": [
         "{\"Key\":\"Name\",\"Value\":\"App1\"}",
         "{\"Key\":\"Owner\",\"Value\":\"Dev\"}"
   ```

```
    ]
  }
}
```

3. Output the RFC template JSON file to a file; this example names it TagCreateAutoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TagCreateAutoRfc.json
```

4. Modify and save the TagCreateAutoRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3cx7we852p3af",
"Title":                "TagCreateAutoRfc"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagCreateAutoRfc.json  --execution-
parameters file://TagCreateAutoParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS infrastructure stacks (stacks named `mc-*`) cannot have tags added with this change type.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3cx7we852p3af.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateTags",
  "Region": "us-east-1",
  "Parameters":     {
    "ResourceArns": [
      "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "arn:aws:ec2:us-east-1:123456789012:volume/vol-1234567890abcdef0",
      "snap-1234567890abcdef0",
      "arn:aws:rds:us-east-1:123456789012:db/my-db-instance",
      "arn:aws:redshift:us-east-1:123456789012:cluster:my-cluster",
      "arn:aws:logs:ap-southeast-2:123456789012:log-group:my-log-group:*"
    ],
    "AddOrUpdateTags": [
      "{\"Key\":\"k1\",\"Value\":\"v1\"}",
      "{\"Key\":\"k2\",\"Value\":\"v2\"}",
      "{\"Key\":\"aws-migration-project-id\",\"Value\":\"project-id\"}"
    ]
  }
}
```

Tag | Create (Review Required)

Add tags to existing, supported resources except those in AMS infrastructure stacks (stacks named mc-*). Tags simplify categorization, identification and targeting AWS resources. For Autoscaling, EC2, Elastic Load Balancing, RDS resources and S3 buckets, use the automated CT ct-3cx7we852p3af.

Full classification: Deployment | Advanced stack components | Tag | Create (review required)

Change Type Details

Change type ID	ct-0176f0n99vcps
Current version	2.0

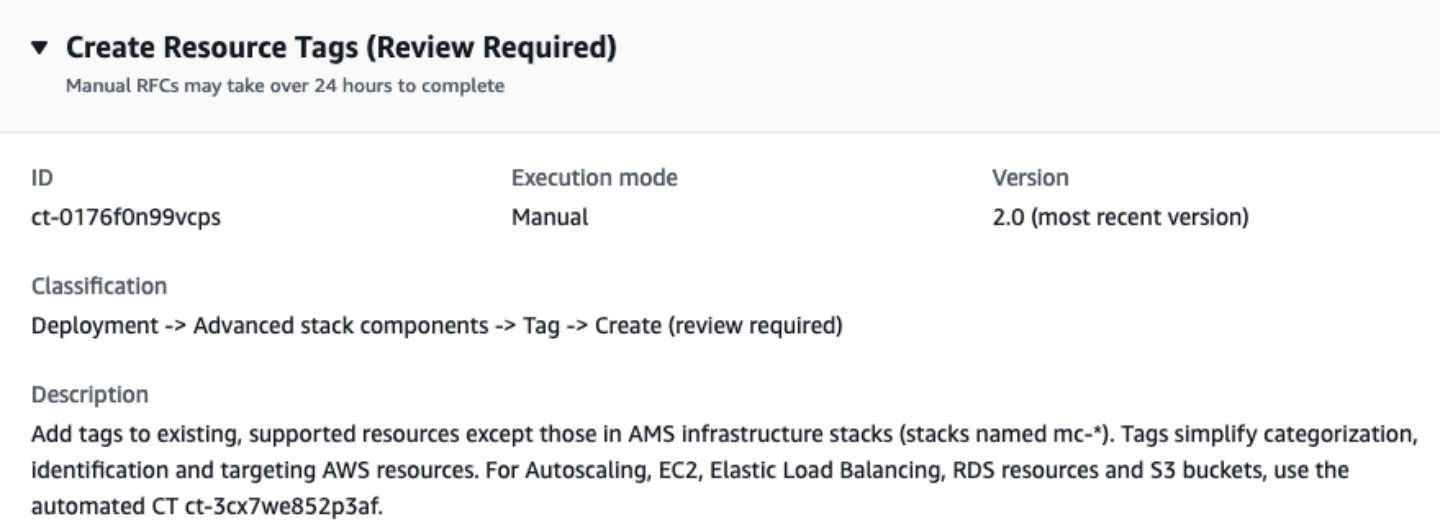
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create tags (review required)

Creating tags (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating tags (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title create-tag --change-type-id ct-0176f0n99vcps --
change-type-version 2.0 --execution-parameters '{"Resources":[{"ResourceArn":"i-
abcd1234","AddOrUpdateTags":[{"Key":"Name","Value":"app-instance-1"},
{"Key":"Owner","Value":"Dep A"}]},{"ResourceArn":"arn:aws:ec2:ap-
southeast-2:123456789012:instance/i-019714a96c22f5452","AddOrUpdateTags":
[{"Key":"Name","Value":"app-instance-2"},{"Key":"Owner","Value":"Dep A"}]}]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagCreateParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagCreateParams.json
   ```

2. Modify and save the TagCreateParams.json file. For example, you can replace the contents with something like this:

   ```
   {
      "Resources": [
       {
         "ResourceArn": "i-abcd1234",
         "AddOrUpdateTags": [
           {
             "Key": "Name",
   ```

```
          "Value": "app-instance-1"
        },
        {
          "Key": "Owner",
          "Value": "Dep A"
        }
      ]
    },
    {
      "ResourceArn": "arn:aws:ec2:ap-southeast-2:123456789012:instance/
i-1234567890abcdef1",
      "AddOrUpdateTags": [
        {
          "Key": "Name",
          "Value": "app-instance-2"
        },
        {
          "Key": "Owner",
          "Value": "Dep A"
        }
      ]
    }
  ]
}
```

3. Output the RFC template JSON file to a file; this example names it TagCreateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TagCreateRfc.json
```

4. Modify and save the TagCreateRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-0176f0n99vcps",
"Title":                "TagCreateRfc"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagCreateRfc.json  --execution-
parameters file://TagCreateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0176f0n99vcps.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Resources": [
    {
      "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "AddOrUpdateTags": [
        {
          "Key": "k1",
          "Value": "v1"
        },
        {
          "Key": "k2",
          "Value": "v2"
        },
        {
          "Key": "k3",
          "Value": "v3"
        }
      ]
    },
    {
      "ResourceArn": "i-0fedcba0987654321",
      "AddOrUpdateTags": [
        {
          "Key": "k1",
          "Value": "v1"
```

```
        },
        {
          "Key": "k2",
          "Value": "v2"
        },
        {
          "Key": "k3",
          "Value": "v3"
        }
      ]
    }
  ],
  "Priority": "Medium"
}
```

Target Group | Create (For ALB)

Use to create a target group for an Application Load Balancer.

Full classification: Deployment | Advanced stack components | Target Group | Create (for ALB)

Change Type Details

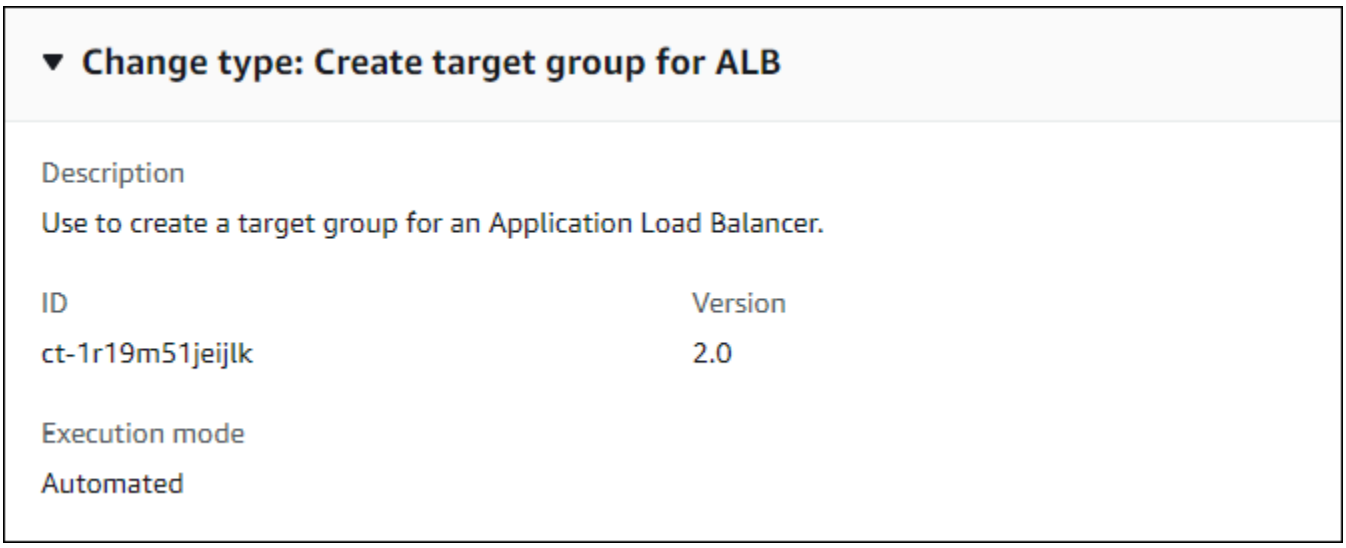
Change type ID	ct-1r19m51jeijlk
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create ALB target group

Creating a Target Group for an Application Load Balancer with the Console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Target Group for an Application Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Version 1.0:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1r19m51jeijlk" --change-type-version "1.0" --title "TITLE" --execution-parameters
 "{\"Description\":\"TargetGroup-ALB\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
 \"stm-9c1t8maqho0os5k21\", \"Name\":\"TG-ALB\", \"TimeoutInMinutes\":60, \"Parameters
\": {\"InstancePort\":\"80\",\"InstanceProtocol\":\"HTTP\"}}"
```

Version 2.0:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-1r19m51jeijlk" --change-type-version "2.0" --title "TITLE" --execution-parameters
 "{\"Description\":\"TargetGroup-ALB\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
 \"stm-9c1t8maqho0os5k22\", \"Name\":\"TG-ALB\", \"TimeoutInMinutes\":60, \"Parameters
\": {\"ApplicationLoadBalancerArn\":\"ARN\",\"InstancePort\":\"80\",\"InstanceProtocol
\":\"HTTP\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateTgAlbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1r19m51jeijlk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateTgAlbParams.json
   ```

2. Modify and save the CreateTgAlbParams file. For example, you can replace the contents with something like this:

   Version 1.0:

   ```
   {
   "Description":      "Target-Group-ALB-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-9c1t8maqho0os5k21",
   "Name":             "My-ALB-Target-Group",

   "Parameters":   {
       "LoadBalancerArn":              ARN,
   ```

```
    "DefaultActionTargetGroupArn":   ARN,
    "Port":                          PORT,
    "Protocol":                     Protocol"
    }
}
```

Version 2.0:

```
{
"Description":      "Target-Group-ALB-Create",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-9c1t8maqho0os5k22",
"Name":             "My-ALB-Target-Group",

"Parameters":   {
    "ApplicationLoadBalancerArn":          ARN,
    "InstancePort":                        PORT,
    "InstanceProtocol":                    Protocol"
    }
}
```

3. Output the RFC template to a file in your current folder named CreateTgAlbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateTgAlbRfc.json
```

4. Modify and save the CreateTgAlbRfc.json file. For example, you can replace the contents with something like this:

Version 1.0:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1r19m51jeijlk",
"Title":                "Target-Group-ALB-Create-RFC"
}
```

Version 2.0:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-1r19m51jeijlk",
"Title":                "Target-Group-ALB-Create-RFC"
```

```
}
```

5. Create the RFC, specifying the CreateTgAlbRfc file and the CreateTgAlbParams file:

```
aws amscm create-rfc --cli-input-json file://CreateTgAlbRfc.json --execution-
parameters file://CreateTgAlbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Next, create a listener for the Application Load Balancer. For more information,see ALB listeners. To open ports and associate security groups, submit a Management | Other | Other | Update change type. For more information, see Other | Other requests.

Tips

> **Note**
>
> The 2.0 version of this change type uses a different StackTemplateId (stm-9c1t8maqho0os5k22) from the 1.0 version. This is important if you are submitting the RFC with this change type version at the command line. The new version includes a new, required, parameter: **ApplicationLoadBalancer**.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1r19m51jeijlk.

Example: Required Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "MyRFCName",
  "StackTemplateId": "stm-9c1t8maqho0os5k22",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "InstancePort": "80",
    "InstanceProtocol": "HTTP",
```

```
    "ApplicationLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-app-load-balancer/abcdefghij"
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "MyRFCName",
  "StackTemplateId": "stm-9c1t8maqho0os5k22",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ApplicationLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/app/my-app-load-balancer/abcdefghij",
    "HealthCheckHealthyThreshold": "5",
    "HealthCheckUnhealthyThreshold": "3",
    "HealthCheckInterval": 30,
    "HealthCheckTimeout": "10",
    "HealthCheckTargetPath": "/healthcheck",
    "HealthCheckTargetPort": "80",
    "HealthCheckTargetProtocol": "HTTP",
    "ValidHTTPCode": "200-259",
    "InstancePort": "80",
    "Name": "mytargetgroup",
    "InstanceProtocol": "HTTP",
    "DeregistrationDelayTimeout": "300",
    "SlowStartDuration": "60",
    "StickinessCookieExpirationPeriod": "3600",
    "TargetType": "ip",
    "Target1ID": "192.168.0.1",
    "Target1Port": "80",
    "Target1AvailabilityZone": "all",
    "Target2ID": "192.168.0.2",
    "Target2Port": "80",
    "Target2AvailabilityZone": "all",
    "Target3ID": "10.44.4.125",
    "Target3Port": "8080",
    "Target3AvailabilityZone": "",
    "Target4ID": "10.44.4.126",
    "Target4Port": "8080",
    "Target4AvailabilityZone": "",
```

```
    "Target5ID": "192.168.0.127",
    "Target5Port": "80",
    "Target5AvailabilityZone": "all",
    "Target6ID": "192.168.0.128",
    "Target6Port": "80",
    "Target6AvailabilityZone": "all",
    "Target7ID": "10.44.4.129",
    "Target7Port": "8080",
    "Target7AvailabilityZone": "",
    "Target8ID": "10.44.4.130",
    "Target8Port": "8080",
    "Target8AvailabilityZone": ""
  }
}
```

Target Group | Create (For NLB)

Use to create a target group for a Network Load Balancer.

Full classification: Deployment | Advanced stack components | Target Group | Create (for NLB)

Change Type Details

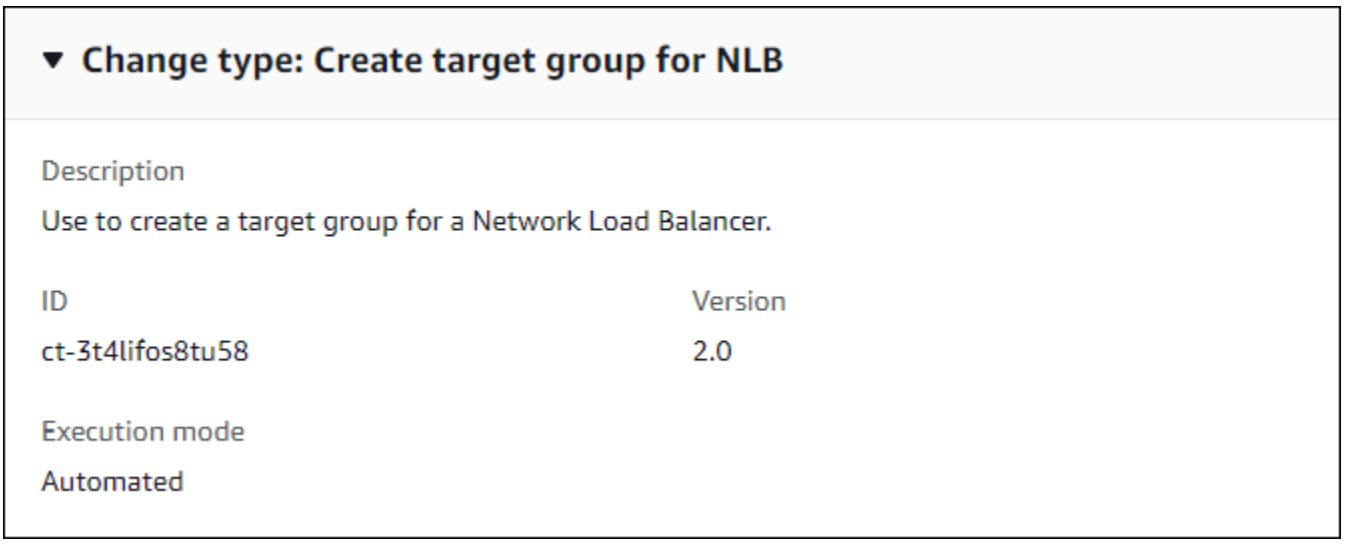
Change type ID	ct-3t4lifos8tu58
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create NLB target group

Creating a Target Group for a Network Load Balancer with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Target Group for a Network Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Version 1.0:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3t4lifos8tu58" --change-type-version "1.0" --title "TITLE" --execution-parameters
 "{\"Description\":\"TargetGroup-NLB\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
 \"stm-6pvp2f7cp481g1r46\", \"Name\":\"TG-NLB\", \"TimeoutInMinutes\":60, \"Parameters
\": {\"InstancePort\":\"80\",\"InstanceProtocol\":\"HTTP\"}}"
```

Version 2.0:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3t4lifos8tu58" --change-type-version "2.0" --title "TITLE" --execution-parameters
 "{\"Description\":\"TargetGroup-NLB\", \"VpcId\":\"VPC_ID\", \"StackTemplateId\":
 \"stm-6pvp2f7cp481g1r47\", \"Name\":\"TG-NLB\", \"TimeoutInMinutes\":60, \"Parameters
\": {\"NetworkLoadBalancerArn\":\"ARN\",\"InstancePort\":\"80\",\"InstanceProtocol\":
\"HTTP\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateTGNlbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3t4lifos8tu58" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateTGNlbParams.json
   ```

2. Modify and save the CreateTGNlbParams file. For example, you can replace the contents with something like this:

   Version 1.0:

   ```
   {
   "Description":      "Target-Group-NLB-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-6pvp2f7cp481g1r46",
   "Name":             "My-NLB-Target-Group",

   "Parameters":   {
   ```

```
    "InstancePort":                          PORT
    }
}
```

Version 2.0:

```
{
"Description":      "Target-Group-NLB-Create",
"VpcId":            "VPC_ID",
"StackTemplateId":  "stm-6pvp2f7cp481g1r47",
"Name":             "My-NLB-Target-Group",

"Parameters":   {
    "NetworkLoadBalancerArn":               ARN,
    "InstancePort":                         PORT
    }
}
```

3. Output the RFC template to a file in your current folder named CreateTgNlbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateTgNlbRfc.json
```

4. Modify and save the CreateTgNlbRfc.json file. For example, you can replace the contents with something like this:

   Version 1.0:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3t4lifos8tu58",
"Title":                "Target-Group-NLB-Create-RFC"
}
```

   Version 2.0:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-3t4lifos8tu58",
"Title":                "Target-Group-NLB-Create-RFC"
}
```

5. Create the RFC, specifying the CreateTgNlbRfc file and the CreateTgNlbParams file:

```
aws amscm create-rfc --cli-input-json file://CreateTgNlbRfc.json --execution-
parameters file://CreateTgNlbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Next Steps: Create a listener for the NLB, see NLB listeners. To open ports and associate security groups, submit a Management | Other | Other | Update change type. For more information, see Other | Other requests.

Tips

> **Important**
>
> There is a new version of this change type, v2.0, that uses a different StackTemplateId (stm-6pvp2f7cp481g1r47). This is important if you are submitting the RFC with this change type at the command line. The new version includes a new, required, parameter: **NetworkLoadBalancer**.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3t4lifos8tu58.

Example: Required Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "MyRFCName",
  "StackTemplateId": "stm-6pvp2f7cp481g1r47",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "InstancePort": "80",
    "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-nlb/abcdefghij"
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "MyRFCName",
  "StackTemplateId": "stm-6pvp2f7cp481g1r47",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "NetworkLoadBalancerArn": "arn:aws:elasticloadbalancing:us-
west-2:123456789012:loadbalancer/net/my-nlb/abcdefghij",
    "HealthCheckHealthyThreshold": "5",
    "HealthCheckInterval": 30,
    "HealthCheckTargetPath": "/healthcheck",
    "HealthCheckTargetPort": "80",
    "HealthCheckTargetProtocol": "HTTP",
    "InstancePort": "80",
    "Name": "mytargetgroup",
    "ProxyProtocolV2": "true",
    "DeregistrationDelayTimeout": "300",
    "TargetType": "ip",
    "Target1ID": "192.168.0.1",
    "Target1Port": "80",
    "Target1AvailabilityZone": "all",
    "Target2ID": "192.168.0.2",
    "Target2Port": "80",
    "Target2AvailabilityZone": "all",
    "Target3ID": "10.44.4.125",
    "Target3Port": "8080",
    "Target3AvailabilityZone": "",
    "Target4ID": "10.44.4.126",
    "Target4Port": "8080",
    "Target4AvailabilityZone": "",
    "Target5ID": "192.168.0.127",
    "Target5Port": "80",
    "Target5AvailabilityZone": "all",
    "Target6ID": "192.168.0.128",
    "Target6Port": "80",
    "Target6AvailabilityZone": "all",
    "Target7ID": "192.168.0.129",
    "Target7Port": "8080",
    "Target7AvailabilityZone": "",
    "Target8ID": "192.168.0.130",
    "Target8Port": "8080",
```

```
    "Target8AvailabilityZone": ""
  }
}
```

VPC | Add Static Route (Review Required)

Create a static route on your route table inside a VPC.

Full classification: Deployment | Advanced stack components | VPC | Add static route (review required)

Change Type Details

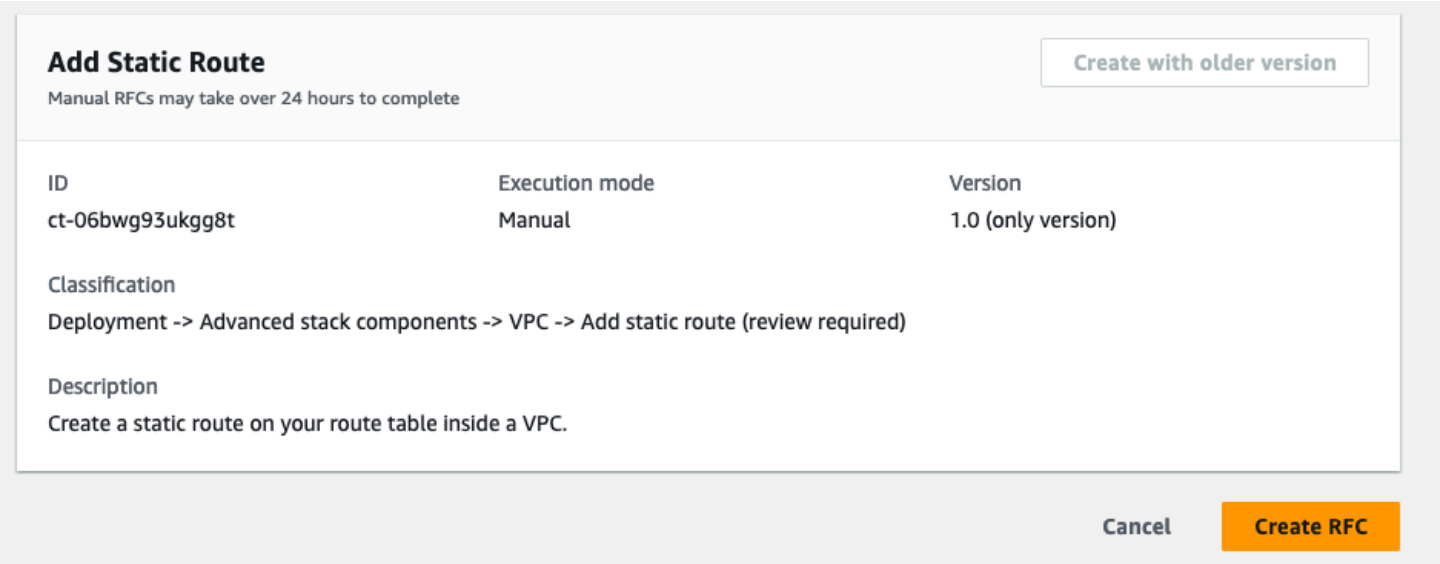
Change type ID	ct-06bwg93ukgg8t
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Add VPC static route (review required)

Add a static route with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a static route with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title="Add Static Route" --description="Add static route"
 --ct-id="ct-06bwg93ukgg8t" --ct-version="1.0" --input-params="{\"RouteTableId
\":\"rtb-0123abcd\",\"DestinationCidrBlock\":\"172.31.0.0/16\",\"Target\":
\"pcx-0123456789abcdefg\",\"Priority\":\"High\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it EncryptAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-06bwg93ukgg8t" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddStaticRouteParams.json
   ```

2. Modify and save the execution AddStaticRouteParams.json file. For example, you can replace the contents with something like this:

   ```
   {
       "RouteTableId": "rtb-0123abcd",
       "DestinationCidrBlock": "172.31.0.0/16",
       "Target": "pcx-0123456789abcdefg",
       "Priority": "High"
   }
   ```

3. Output the RFC template JSON file; this example names it AddStaticRouteRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AddStaticRouteRfc.json
   ```

4. Modify and save the AddStaticRouteRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-06bwg93ukgg8t",
     "Title": "Add static route"
   }
   ```

5. Create the RFC, specifying the AddStaticRouteRfc file and the AddStaticRouteParams file:

```
aws amscm create-rfc --cli-input-json file://AddStaticRouteRfc.json  --execution-
parameters file://AddStaticRouteParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about VPCs, see Virtual private clouds (VPC).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-06bwg93ukgg8t.

Example: Required Parameters

```
{
    "RouteTableId": "rtb-01234567890abcdef",
    "Destination": "192.168.10.0/24",
    "RouteTableTarget": "igw-01234567890abcdef"
}
```

Example: All Parameters

```
{
    "RouteTableId": "rtb-01234567890abcdef",
    "Destination": "192.168.10.0/24",
    "RouteTableTarget": "igw-01234567890abcdef",
    "Priority": "High"
}
```

VPC Endpoint (Interface) | Create

Create an interface VPC endpoint, which allows you to connect to services powered by AWS PrivateLink, including many AWS services.

Full classification: Deployment | Advanced stack components | VPC endpoint (interface) | Create

Change Type Details

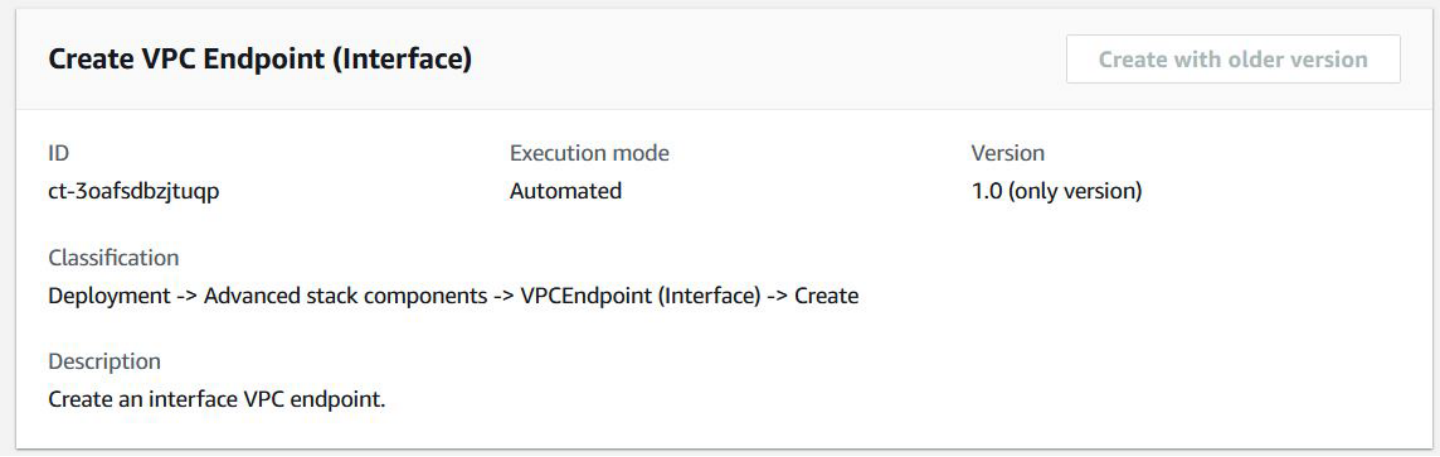
Change type ID	ct-3oafsdbzjtuqp
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create VPC endpoint

Creating a VPC endpoint with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a VPC endpoint with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3oafsdbzjtuqp" --change-type-version
 "1.0" --title "Create VPC Endpoint" --execution-parameters "{\"Description\":\"VPC
 endpoint interface\",\"VpcId\":\"vpc-1234567890abcdef0\",\"Name\":\"VPC endpoint
 interface\",\"StackTemplateId\":\"stm-f0cumpt1rfc1p1739\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"VpcId\":\"vpc-1234567890abcdef0\",\"ServiceName\":
\"com.amazonaws.us-east-1.codedeploy\",\"SecurityGroups\":[\"sg-1234567890abcdef0\",
\"sg-1234567890abcdef1\"],\"SubnetIds\":[\"subnet-1234567890abcdef0\",
\"subnet-1234567890abcdef1\"],\"EnablePrivateDns\":\"false\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it VPCEndpointCreateParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3oafsdbzjtuqp"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    VPCEndpointCreateParams.json
   ```

2. Modify and save the execution parameters as VPCEndpointCreateParams.json. For example, you can replace the contents with something like this:

```
{
  "Description": "VPC endpoint interface",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "VPC endpoint interface",
  "StackTemplateId": "stm-f0cumpt1rfc1p1739",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "VpcId": "vpc-1234567890abcdef0",
    "ServiceName": "com.amazonaws.us-east-1.codedeploy",
    "SecurityGroups": [
      "sg-1234567890abcdef0",
      "sg-1234567890abcdef1"
    ],
    "SubnetIds": [
      "subnet-1234567890abcdef0",
      "subnet-1234567890abcdef1"
    ],
    "EnablePrivateDns": "false"
  }
}
```

3. Output the RFC template JSON file; this example names it VPCEndpointCreateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > VPCEndpointCreateRfc.json
```

4. Modify and save the VPNGatewayCreateRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion" : "1.0",
  "ChangeTypeId" : "ct-3oafsdbzjtuqp",
  "Title" : "Create VPC Endpoint "
}
```

5. Create the RFC, specifying the VPCEndpointCreateRfc file and the VPCEndpointCreateParams file:

```
aws amscm create-rfc --cli-input-json file://VPCEndpointCreateRfc.json  --
execution-parameters file://VPCEndpointCreateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3oafsdbzjtuqp.

Example: Required Parameters

```
{
  "Description": "Test description",
  "VpcId": "vpc-12345678901234567",
  "Name": "TestStack",
  "StackTemplateId": "stm-f0cumpt1rfc1p1739",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "VpcId": "vpc-a388bbc4",
    "ServiceName": "com.amazonaws.ap-southeast-2.execute-api",
    "SecurityGroups": [
      "sg-94bdebea",
      "sg-007442bef5c5badff"
    ],
    "SubnetIds": [
      "subnet-5b706a3c",
      "subnet-c9809480"
    ]
  }
}
```

Example: All Parameters

```
{
  "Description": "Test description",
  "VpcId": "vpc-12345678",
  "Name": "TestStack",
  "Tags": [
    {
      "Key": "foo",
```

```
      "Value": "bar"
    }
  ],
  "StackTemplateId": "stm-f0cumpt1rfc1p1739",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "VpcId": "vpc-a388bbc4",
    "ServiceName": "com.amazonaws.ap-southeast-2.execute-api",
    "SecurityGroups": [
      "sg-94bdebea",
      "sg-007442be"
    ],
    "SubnetIds": [
      "subnet-5b706a3c",
      "subnet-c9809480"
    ],
    "EnablePrivateDns": "true"
  }
}
```

VPN Gateway | Create

Create a virtual private network (VPN) gateway (the endpoint on the VPC side of your VPN connection), and associate it to an existing virtual private cloud (VPC) in your account.

Full classification: Deployment | Advanced stack components | VPN Gateway | Create

Change Type Details

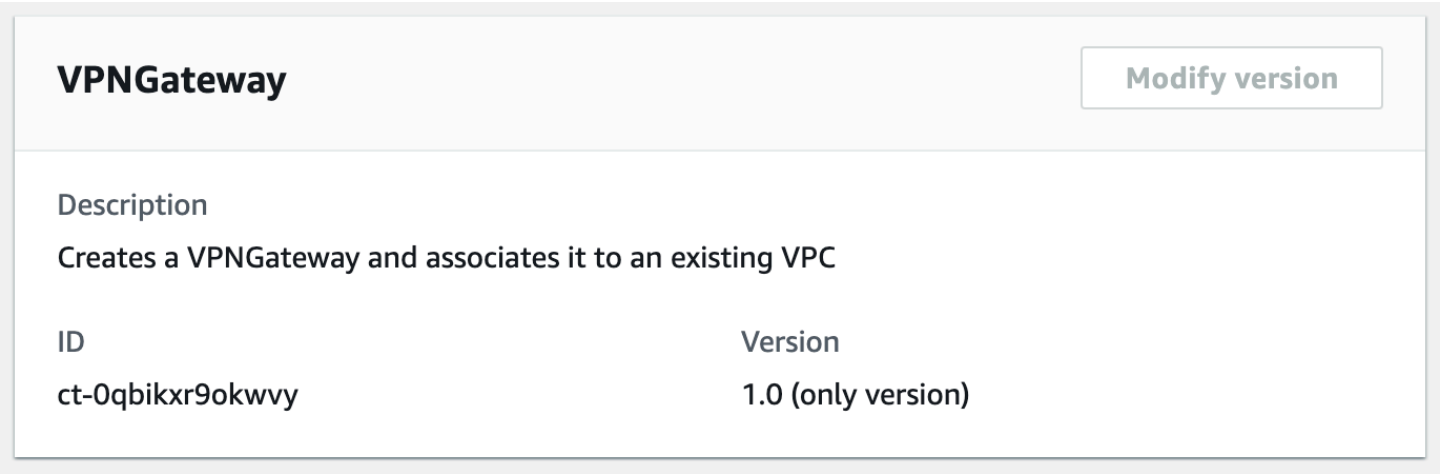
Change type ID	ct-0qbikxr9okwvy
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create VPN gateway

Creating a VPN gateway with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a VPN gateway with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title create-rds-db-instance-point-in-time-
restore --change-type-id ct-0qbikxr9okwvy --change-type-version
 1.0 --execution-parameters '{"Description": "test","VpcId":
 "vpc-317a9856","Name": "newvpngateway","StackTemplateId": "stm-
mcti3bha1vhon1sie","TimeoutInMinutes": 60,"Parameters": {"VpcId":
 "vpc-317a9856","AmazonSideAsn": 64512,"Name": "test"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it EncryptAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0qbikxr9okwvy"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    VPNGatewayCreateParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "test",
     "VpcId": "vpc-317a9856",
     "Name": "newvpngateway",
     "StackTemplateId": "stm-mcti3bha1vhon1sie",
     "TimeoutInMinutes": 60,
     "Parameters": {
       "VpcId": "vpc-317a9856",
       "AmazonSideAsn": 64512,
       "Name": "test"
     }
   }
   ```

3. Output the RFC template JSON file; this example names it VPNGatewayCreateRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > VPNGatewayCreateRfc.json
   ```

4. Modify and save the VPNGatewayCreateRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion" : "1.0",
  "ChangeTypeId" : "ct-0qbikxr9okwvy",
  "Title" : "VPNGateway Create"
}
```

5. Create the RFC, specifying the VPNGatewayCreateRfc file and the VPNGatewayCreateParams file:

```
aws amscm create-rfc --cli-input-json file://VPNGatewayCreateRfc.json  --execution-
parameters file://VPNGatewayCreateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0qbikxr9okwvy.

Example: Required Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-mcti3bha1vhon1sie",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "VpcId" : "vpc-12345678901234567"
  }
}
```

Example: All Parameters

```
{
```

```
  "Description" : "Test description",
  "VpcId" : "vpc-12345678",
  "Name" : "TestStack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-mcti3bha1vhon1sie",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "VpcId" : "vpc-12345678",
    "AmazonSideAsn" : 64512,
    "Name" : "Test-VPNGateway"
  }
}
```

AMS Patterns Subcategory

Change Type Items and Operations in the AMS Patterns Subcategory

- Solution | Deploy (Review Required)

Solution | Deploy (Review Required)

Deploy an AMS pattern to the current account. Patterns provide tools, architectures, and step-by-step guidance for implementing the methodologies for the migration strategy. Multi-account landing zone accounts can also specify OrganizationalUnit to deploy the pattern to all the accounts in that OU.

Full classification: Deployment | AMS patterns | Solution | Deploy (review required)

Change Type Details

Change type ID	ct-2jndrh7uit8uf
Current version	1.0
Expected execution duration	60 minutes

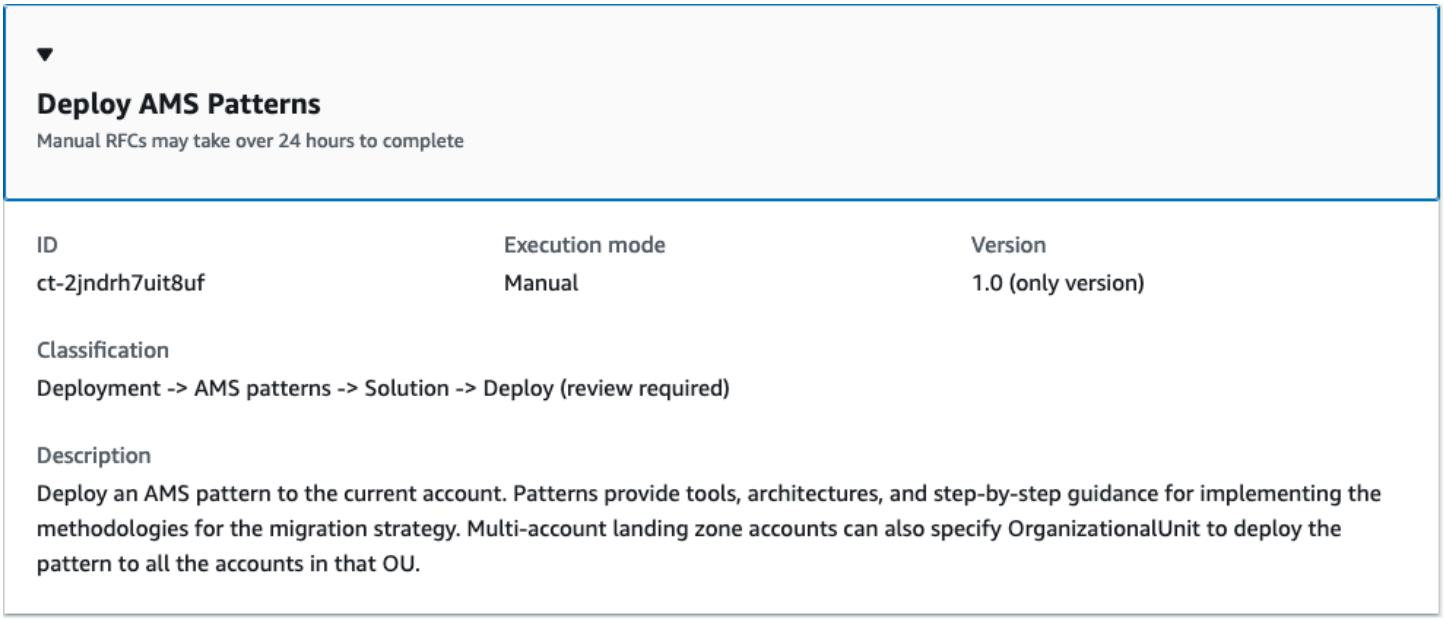
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Deploy AMS Pattern (review required)

Creating an AWS KMS Key (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AWS KMS Key (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2jndrh7uit8uf" --change-type-version
 "1.0" --title "Deploy AMS Patterns" --execution-parameters "{\"PatternName
\": \"amsEbsVolumeSnapshotTagger\",\"PatternParameters\": "{\"ExcludedTags\":
\"BackupProd,Backup\",\"ASMGuardRail\":\"enabled\"}", \"OrganizationalUnit\":
 \"ou-9dyd-s2vptest\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it DeployAMSPatternsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2jndrh7uit8uf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeployAMSPatternsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "PatternName": "amsEbsVolumeSnapshotTagger",
       "Parameters": {
           "ExcludedTags": "BackupProd,Backup",
           "ASMGuardRail": "enabled"
       },
   ```

```
    "OrganizationalUnit": "ou-9dyd-s2vptest"
}
```

3. Output the RFC template JSON file; this example names it DeployAMSPatternsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployAMSPatternsRfc.json
```

4. Modify and save the DeployAMSPatternsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2jndrh7uit8uf",
"Title":                "Deploy AMS Patterns"
}
```

5. Create the RFC, specifying the DeployAMSPatternsRfc file and the DeployAMSPatternsParams file:

```
aws amscm create-rfc --cli-input-json file://DeployAMSPatternsRfc.json --execution-
parameters file://DeployAMSPatternsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2jndrh7uit8uf.

Example: Required Parameters

```
{
  "PatternName": "amsEbsVolumeSnapshotTagger"
}
```

Example: All Parameters

```
{
  "PatternName": "amsEbsVolumeSnapshotTagger",
```

```
  "OrganizationalUnitIds": ["ou-9dyd-jvsei4yg"],
  "Priority": "Medium",
  "PatternParameters": [
    {
      "Name": "Foo",
      "Value": "Bar"
    }
  ]
}
```

AMS Resource Scheduler Subcategory

Change Type Items and Operations in the AMS Resource Scheduler Subcategory

- Solution | Deploy

Solution | Deploy

Deploy the AMS Resource Scheduler solution in the account. The AMS Resource Scheduler lets you schedule automatic start and/or stop for Auto Scaling groups, EC2s, and RDS instances. Note that the Resource Scheduler deploys in an enabled state, by default; you can manage that with the AMS Resource Scheduler Disable and Enable change types.

Full classification: Deployment | AMS Resource Scheduler | Solution | Deploy

Change Type Details

Change type ID	ct-0ywnhc8e5k9z5
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Deploy AMS Resource Scheduler Solution

Deploying AMS Resource Scheduler solution with the console

The following shows this change type in the AMS console.

Deploy AMS Resource Scheduler

ID	Execution mode	Version
ct-0ywnhc8e5k9z5	Automated	2.0 (most recent version)

Classification

Deployment -> AMS Resource Scheduler -> Solution -> Deploy

Description

Deploy the AMS Resource Scheduler solution in the account. The AMS Resource Scheduler lets you schedule automatic start and/or stop for Auto Scaling groups, EC2s, and RDS instances. Note that the Resource Scheduler deploys in an enabled state, by default; you can manage that with the AMS Resource Scheduler Disable and Enable change types.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deploying AMS Resource Scheduler solution with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id ct-0ywnhc8e5k9z5 --change-type-
version "2.0" --title "Deploy Resource Scheduler" --execution-parameters
 '{"DocumentName":"AWSManagedServices-HandleAMSResourceSchedulerStack-
Admin","Region":"us-east-1","Parameters":{"SchedulingActive":
["Yes"],"ScheduledServices":["ec2,rds,autoscaling"],"TagName":
["Schedule"],"DefaultTimezone":["America/New_York"],"Action":["Deploy"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DeployResSchedulerParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ywnhc8e5k9z5"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeployResSchedulerParams.json
   ```

2. Modify and save the DeployResSchedulerParams file.

   ```
   {
     "DocumentName": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "SchedulingActive": [
         "Yes"
       ],
       "ScheduledServices": [
         "ec2,rds,autoscaling"
       ],
       "TagName": [
         "Schedule"
       ],
       "DefaultTimezone": [
         "America/New_York"
       ],
       "Action": [
         "Deploy"
   ```

```
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it DeployResSchedulerRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployResSchedulerRfc.json
```

4. Modify and save the DeployResSchedulerRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-0ywnhc8e5k9z5",
"Title":                "Deploy AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the DeployResSchedulerRfc file and the DeployResSchedulerParams file:

```
aws amscm create-rfc --cli-input-json file://DeployResSchedulerRfc.json --
execution-parameters file://DeployResSchedulerParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For background information, see How the AMS Resource Scheduler works. For a quick-start tutorial, see AMS Resource Scheduler quick start.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ywnhc8e5k9z5.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["Deploy"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SchedulingActive" : [
      "Yes"
    ],
    "ScheduledServices" : [
      "ec2,rds,autoscaling"
    ],
    "TagName" : [
      "MySchedule"
    ],
    "DefaultTimezone" : [
      "Australia/Sydney"
    ],
    "UseCMK" : [
      "arn:aws:kms:ap-southeast-1:830123456789:key/07aaab3c-50d3-4cd8-
ab61-3de57127dab9"
    ],
    "UseLicenseManager" : [
      "arn:aws:license-manager:ap-southeast-1:830123456789:license-
configuration:lic-78c1e0cfc1233a4eac7197d7ee57f92c"
    ],
    "Action": [
      "Deploy"
    ]
  }
}
```

Applications Subcategory

Change Type Items and Operations in the Applications Subcategory

CodeDeploy Application | Create

Use to create an AWS CodeDeploy application resource with the specified name.

Full classification: Deployment | Applications | CodeDeploy application | Create

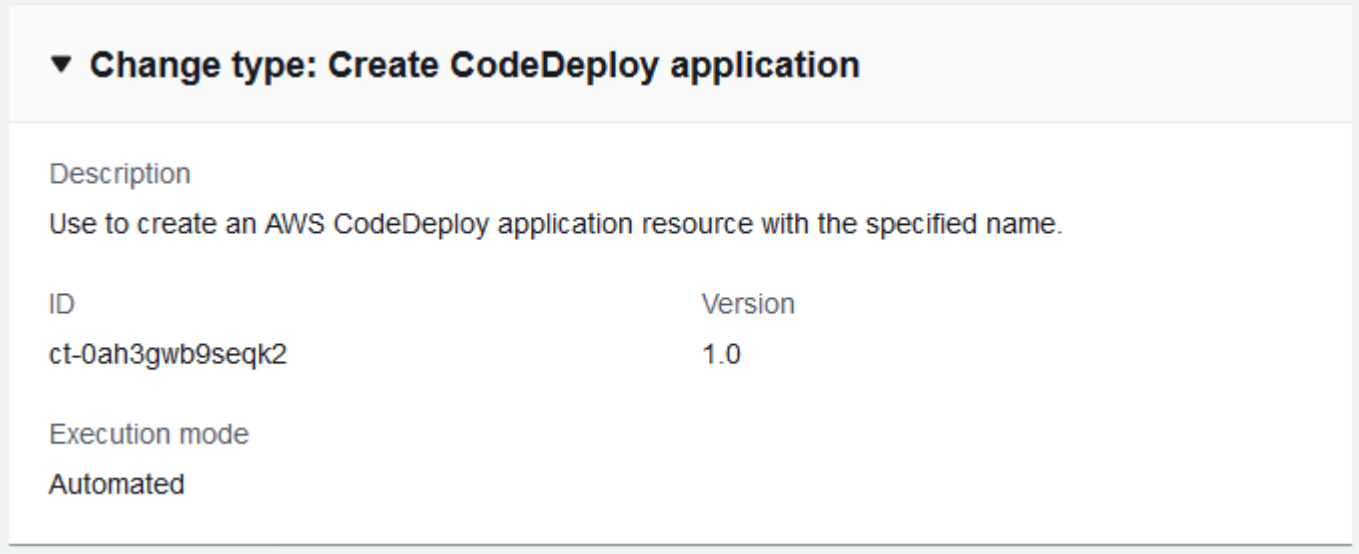
Change Type Details

Change type ID	ct-0ah3gwb9seqk2
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CodeDeploy application

Creating a CodeDeploy application with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CodeDeploy application with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0ah3gwb9seqk2" --change-type-version "1.0"
 --title "Stack-Create-CD-App" --execution-parameters "{\"Description\":\"TestCdApp\",
\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-sft6rv00000000000\",\"Name\":\"Test\",
\"TimeoutInMinutes\":60,\"Parameters\":{\"CodeDeployApplicationName\":\"Test\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for the CodeDeploy application CT to a file in your current folder; this example names it CreateCDAppParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ah3gwb9seqk2" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateCDAppParams.json
   ```

2. Modify and save the JSON file as follows. For example, you can replace the contents with something like this:

   ```
   {
   "Description":                    "Create WP CodeDeploy App",
   "VpcId":                          "VPC_ID",
   "StackTemplateId":                "stm-sft6rv00000000000",
   "Name":                           "WpCDApp",
   "TimeoutInMinutes":               60,
   "Parameters":   {
       "CodeDeployApplicationName":    "WordPressCDApp"
       }
   }
   ```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it CreateCDAppRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateCDAppRfc.json
   ```

4. Modify and save the JSON file as follows. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0ah3gwb9seqk2",
"Title":                "CD-App-Stack-RFC"
}
```

5. Create the RFC, specifying the CreateCDAppRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateCDAppRfc.json --execution-
parameters file://CreateCDAppParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS CodeDeploy, see Create an Application with AWS CodeDeploy.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ah3gwb9seqk2.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-sft6rv00000000000",
  "Name": "Name your stack",
  "Tags": [{"Key":"foo","Value":"bar"}],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CodeDeployApplicationName": "foobarapp"
  }
```

```
}
```

CodeDeploy Application | Deploy

Deploy a revision of an existing AWS CodeDeploy application, which are source files CodeDeploy will deploy to your instances or scripts CodeDeploy will run on your instances.

Full classification: Deployment | Applications | CodeDeploy application | Deploy

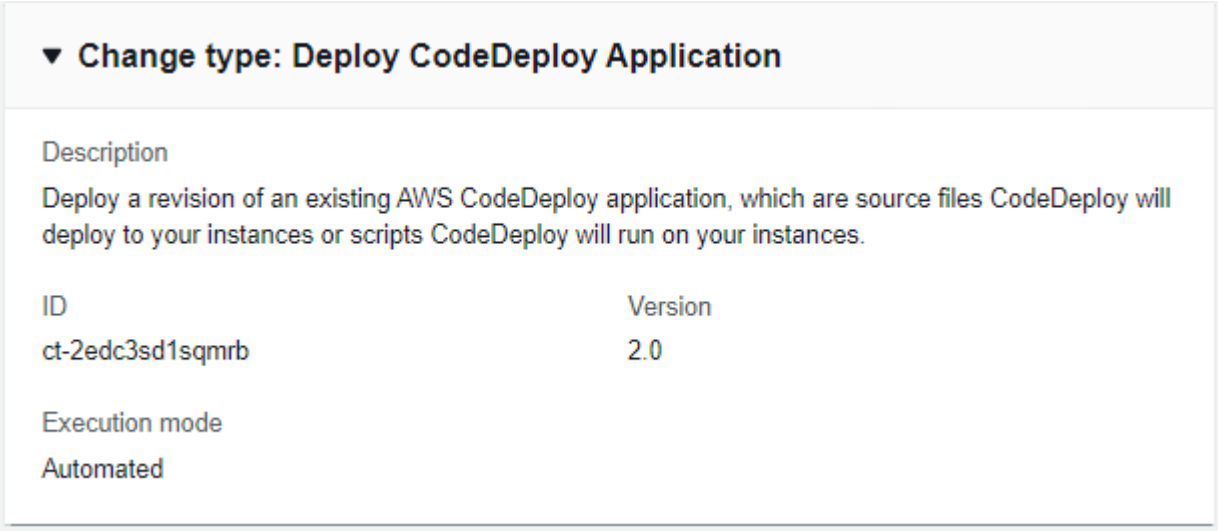
Change Type Details

Change type ID	ct-2edc3sd1sqmrb
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Deploy CodeDeploy application

Deploying a CodeDeploy application with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deploying a CodeDeploy application with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2edc3sd1sqmrb" --change-
type-version "2.0" --title "Stack-Deploy-CD-App" --execution-
parameters "{\"Description\":\"MyCDAppDeployTest\",\"VpcId\":
\"VPC_ID\",\"Name\":\"Test\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployDeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"CodeDeployDeploymentGroupName\":\"TestCDDepGroup\",
\"CodeDeployIgnoreApplicationStopFailures\":false,\"CodeDeployRevision\":
{\"RevisionType\":\"S3\",\"S3Location\":{\"S3Bucket\":\"TestBucket\",\"S3BundleType\":
\"tar\",\"S3Key\":\"TestKey\"}}}}"Test\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for the CodeDeploy application deployment CT; this example names it DeployCDAppParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-2edc3sd1sqmrb" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > DeployCDAppParams.json
```

2. Modify the JSON file as follows. For example, you can replace the contents with something like this:

```
{
"Description":                      "Deploy WordPress CodeDeploy Application",
"VpcId":                            "VPC_ID",
"Name":                             "WP CodeDeploy Deployment Group",
"TimeoutInMinutes":                 360,
"Parameters":   {
    "CodeDeployApplicationName":        "WordPressCDApp",
    "CodeDeployDeploymentGroupName":    "WordPressCDDepGroup",
    "CodeDeployIgnoreApplicationStopFailures": false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "ACCOUNT_ID.BUCKET_NAME",
        "S3BundleType": "zip",
        "S3Key": "wordpress.zip" }
        }
    }
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it DeployCDAppRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeployCDAppRfc.json
```

4. Modify and save the DeployCDAppRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2edc3sd1sqmrb",
"Title":                "CD-Deploy-For-CD-APP-Stack-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the DeployCDAppRfc file:

```
aws amscm create-rfc --cli-input-json file://DeployCDAppRfc.json  --execution-
parameters file://DeployCDAppParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see Create a deployment with CodeDeploy.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2edc3sd1sqmrb.

Example: Required Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-01234567890abcdef",
  "Name": "Name your stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CodeDeployApplicationName": "foobarapp",
    "CodeDeployDeploymentGroupName": "myfoogroup",
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "mybucket",
        "S3BundleType": "zip",
        "S3Key": "mykey"
      }
    }
  }
}
```

Example: All Parameters

```
{
  "Description": "Stack Description.",
```

```
  "VpcId": "vpc-12345678",
  "Name": "Name your stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CodeDeployApplicationName": "foobarapp",
    "CodeDeployDeploymentConfigName": "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName": "myfoogroup",
    "CodeDeployIgnoreApplicationStopFailures": false,
    "CodeDeployRevision": {
      "RevisionType": "S3",
      "S3Location": {
        "S3Bucket": "mybucket",
        "S3BundleType": "zip",
        "S3ETag": "1234567",
        "S3Key": "mykey",
        "S3Version": "versionfoo"
      }
    }
  }
}
```

CodeDeploy Deployment Group | Create

Use to create an AWS CodeDeploy application deployment group, an entity that describes what instances to deploy a given application to.

Full classification: Deployment | Applications | CodeDeploy deployment group | Create

Change Type Details

Change type ID	ct-2gd0u847qd9d2
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CodeDeploy deployment group

Creating a CodeDeploy deployment group with the console

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CodeDeploy deployment group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2gd0u847qd9d2" --change-type-version
 "1.0" --title "Stack-Create-CD-Dep-Group" --execution-parameters "{\"Description
\":\"TestCdDepGroupRfc\",\"VpcId\":\"VPC_ID\",\"StackTemplateId\":\"stm-
sp9lrk00000000000\",\"Name\":\"MyTestCDDepGroup\",\"TimeoutInMinutes\":60,\"Parameters
\":{\"CodeDeployApplicationName\":\"TestCDApp\",\"CodeDeployAutoScalingGroups\":
[\"TestASG\"],\"CodeDeployDeploymentConfigName\":\"CodeDeployDefault.OneAtATime\",
\"CodeDeployDeploymentGroupName\":\"Test\",\"CodeDeployServiceRoleArn\":
\"arn:aws:iam::000000000:role/aws-codedeploy-role\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder; this example names it CreateCDDepGroupParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2gd0u847qd9d2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateCDDepGroupParams.json
   ```

2. Modify and save the JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":                    "CreateCDDeploymentGroup",
   "VpcId":                          "VPC_ID",
   "StackTemplateId":                "stm-sp9lrk00000000000",
   "Name":                           "WordPressCDAppGroup",
   "TimeoutInMinutes":               60,
   "Parameters":   {
       "CodeDeployApplicationName":        "WordPressCDApp",
       "CodeDeployAutoScalingGroups":      ["ASG_NAME"],
       "CodeDeployDeploymentConfigName":   "CodeDeployDefault.HalfAtATime",
       "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
       "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
   codedeploy-role"
       }
   }
   ```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it CreateCDDepGroupRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupRfc.json
```

4. Modify and save the JSON file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2gd0u847qd9d2",
"Title":                "CD-Dep-Group-RFC"
}
```

5. Create the RFC, specifying the CreateCDDepGroupRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupRfc.json --execution-
parameters file://CreateCDDepGroupParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS CodeDeploy deployment groups, see Create a Deployment Group with AWS CodeDeploy.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2gd0u847qd9d2.

Example: Required Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-01234567890abcdef",
  "StackTemplateId": "stm-sp9lrk00000000000",
  "Name": "Name your stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
```

```
    "CodeDeployApplicationName": "foobarapp",
    "CodeDeployAutoScalingGroups": ["myfooasg"],
    "CodeDeployDeploymentGroupName": "mydeploymentgroup",
    "CodeDeployServiceRoleArn": "foo::arn::bar"
  }
}
```

Example: All Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-01234567890abcdef",
  "StackTemplateId": "stm-sp9lrk00000000000",
  "Name": "Name your stack",
  "Tags": [{"Key":"foo","Value":"bar"}],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "CodeDeployApplicationName": "foobarapp",
    "CodeDeployAutoScalingGroups": ["myfooasg"],
    "CodeDeployDeploymentConfigName": "CodeDeployDefault.HalfAtATime",
    "CodeDeployDeploymentGroupName": "mydeploymentgroup",
    "CodeDeployServiceRoleArn": "foo::arn::bar"
  }
}
```

CodeDeploy Deployment Group | Create (For EC2 Instance)

Create an AWS CodeDeploy application deployment group specifically for an EC2 instance as target. Tags you create in the EC2 instances, and specify here (EC2FilterTag1, 2, and 3), mark the instances as targets for the deployment group. A name for the deployment group is automatically generated.

Full classification: Deployment | Applications | CodeDeploy deployment group | Create (for EC2 instance)

Change Type Details

Change type ID	ct-00tlkda4242x7
Current version	1.0

Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CodeDeploy deployment group for EC2

Creating a CodeDeploy deployment group for EC2 with the console

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CodeDeploy deployment group for EC2 with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-00tlkda4242x7" --change-type-
version "1.0" --title "Stack-Create-CD-Ec2-Dep-Group" --execution-parameters
 "{\"Description\":\"MyTestCdDepEc2DepGroup\",\"VpcId\":\"VPC_ID\",\"Name\":
\"TestCDDepEc2Group\",\"StackTemplateId\":\"stm-n3hsoirgqeqqdbpk2\",\"TimeoutInMinutes
\":60,\"Parameters\":{\"ApplicationName\":\"TestCDApp\",\"DeploymentConfigName\":
\"CodeDeployDefault.OneAtATime\",\"AutoRollbackEnabled\":\"False\",\"EC2FilterTag\":
\"Name=Test\",\"EC2FilterTag2\":\"\",\"EC2FilterTag3\":\"\",\"ServiceRoleArn\":\"\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file; this example names it CreateCDDepGroupEc2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-00tlkda4242x7"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateCDDepGroupEc2Params.json
   ```

2. Modify and save the JSON file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"Description":                          "CreateCDDepGroupEc2",
"VpcId":                                "VPC_ID",
"StackTemplateId":                      "stm-n3hsoirgqeqqdbpk2",
"Name":                                 "CDAppGroupEc2",
"TimeoutInMinutes":                     60,
"Parameters":   {
    "ApplicationName":          "CDAppEc2",
    "DeploymentConfigName":     "CodeDeployDefault.OneAtATime",
    "CodeDeployDeploymentGroupName":    "UNIQUE_CDDepGroupNAME",
    "CodeDeployServiceRoleArn":         "arn:aws:iam::ACCOUNT_ID:role/aws-
codedeploy-role"
    }
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it CreateCDDepGroupEc2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCDDepGroupEc2Rfc.json
```

4. Modify and save the JSON file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-00tlkda4242x7",
"Title":                "CD-Dep-Group-For-Ec2-Stack-RFC"
}
```

5. Create the RFC, specifying the CreateCDDepGroupEc2Rfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateCDDepGroupEc2Rfc.json --
execution-parameters file://CreateCDDepGroupEc2Params.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS CodeDeploy deployment groups, see Create a Deployment Group with AWS CodeDeploy.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-00tlkda4242x7.

Example: Required Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-01234567890abcdef",
  "StackTemplateId": "stm-n3hsoirgqeqqdbpk2",
  "Name": "Name your stack",
  "Tags": [{"Key":"foo","Value":"bar"}],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ApplicationName": "foobarapp",
    "EC2FilterTag": "Key1=Value1"
  }
}
```

Example: All Parameters

```
{
  "Description": "Stack Description.",
  "VpcId": "vpc-01234567890abcdef",
  "StackTemplateId": "stm-n3hsoirgqeqqdbpk2",
  "Name": "Name your stack",
  "Tags": [{"Key":"foo","Value":"bar"}],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "ApplicationName": "foobarapp",
    "DeploymentConfigName": "CodeDeployDefault.HalfAtATime",
    "AutoRollbackEnabled": "True",
    "EC2FilterTag": "Key1=Value1",
    "EC2FilterTag2": "Key2=Value2",
    "EC2FilterTag3": "Key3=Value3",
    "ServiceRoleArn": "arn:aws:iam::123456789012:role/test02"
  }
}
```

AWS Backup Subcategory

Change Type Items and Operations in the AWS Backup Subcategory

- Backup Plan | Create

Backup Plan | Create

Create an AWS Backup plan, a policy expression that defines when and how you want to back up your AWS resources.

Full classification: Deployment | AWS Backup | Backup plan | Create

Change Type Details

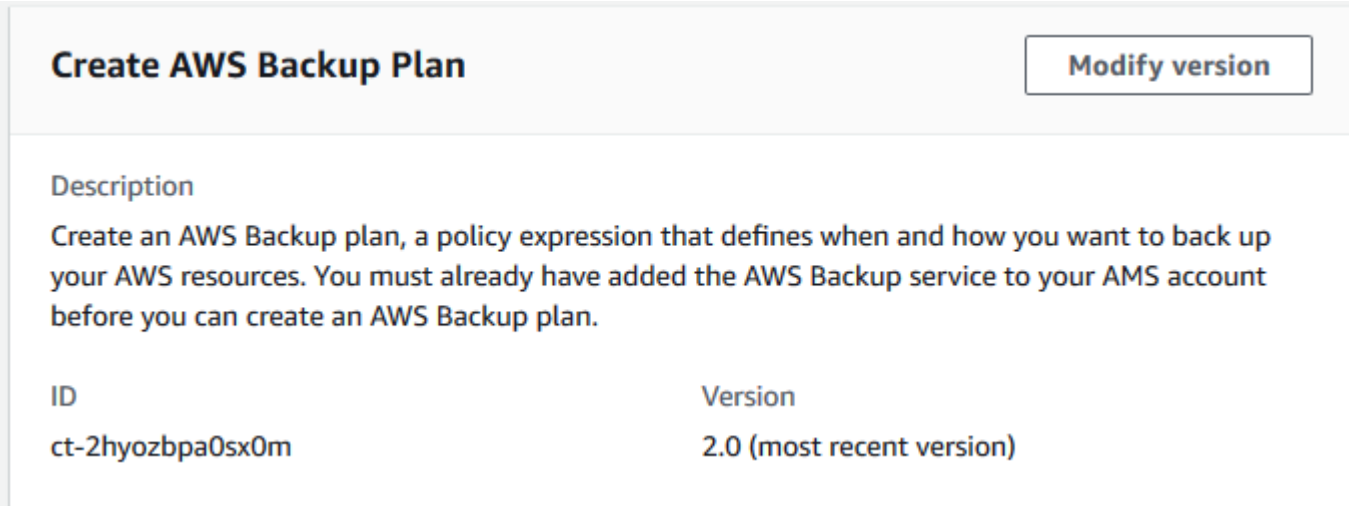
Change type ID	ct-2hyozbpa0sx0m
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create AWS Backup plan

Creating an AWS Backup plan with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an AWS Backup plan with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

With all parameters for one rule:

```
aws amscm create-rfc --change-type-id "ct-2hyozbpa0sx0m" --change-type-version "2.0" --
title "AWS Backup custom backup plan for AMS" \
  --description "RFC_DESCRIPTION" \
  --execution-parameters "{\"VpcId\":\"VPC_ID\",\"Description\":\"PLAN_DESCRIPTION\",
\"Parameters\":{\"BackupPlanName\":\"PLAN_NAME\",\"ResourceTagKey\":\"TAG_KEY\",
\"ResourceTagValue\":\"TAG_VALUE\",\"BackupRule1Name\":\"RULE_NAME\",
\"BackupRule1Vault\":\"VAULT\",\"BackupRule1CompletionWindowMinutes
\":120,\"BackupRule1ScheduleExpression\":\"cron(0 1 ? * * *)\",
\"BackupRule1DeleteAfterDays\":90,\"BackupRule1MoveToColdStorageAfterDays\":365,
\"BackupRule1StartWindowMinutes\":60,\"BackupRule1RecoveryPointTagKey\":\"TAG_KEY\",
\"BackupRule1RecoveryPointTagValue\":\"TAG_VALUE\,\"BackupRule1EnableContinuousBackup
\":\"false\",\"BackupRule1CopyActionsDestVaultArn\":\"VAULT\",
\"BackupRule1CAMoveToColdStorageAfterDays\":0,\"BackupRule1CopyActionsDeleteAfterDays
\":90},\"StackTemplateId\":\"stm-sc68a620000000000\",\"TimeoutInMinutes\":60,\"Name\":
\"TEST_STACK\"}"
```

With only required parameters for one rule:

```
aws amscm create-rfc --change-type-id "ct-2hyozbpa0sx0m" --change-type-version "2.0" --
title "AWS Backup custom backup plan for AMS" \
  --description "RFC_DESCRIPTION" \
  --execution-parameters "{\"VpcId\":\"VPC_ID\",\"Description\":\"PLAN_DESCRIPTION\",
\"Parameters\":{\"BackupPlanName\":\"PLAN_NAME\",\"ResourceTagKey\":\"TAG_KEY\",
\"ResourceTagValue\":\"TAG_VALUE\",\"BackupRule1Name\":\"RULE_NAME\",\"BackupRule1Vault
\":\"VAULT\",\"BackupRule1ScheduleExpression\":\"cron(0 1 ? * * *)\"},\"StackTemplateId
\":\"stm-sc68a620000000000\",\"TimeoutInMinutes\":60,\"Name\":\"TEST_STACK\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateBackupPlanParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hyozbpa0sx0m"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateBackupPlanParams.json
   ```

2. Modify and save the CreateBackupPlanParams file. Note that the **BackupRule1EnableContinuousBackup** parameter set to **true** causes AWS Backup to create continuous backups capable of point-in-time restore (PITR); the default for this parameter is **false**.

```
{
    "VpcId": "VPC_ID",
    "Description": "PLAN_DESCRIPTION",
    "Parameters": {
      "BackupPlanName" : "PLAN_NAME",
      "ResourceTagKey" : "TAG_KEY",
      "ResourceTagValue" : "TAG_VALUE",
      "BackupRule1Name" : "RULE_NAME",
      "BackupRule1Vault" : "VAULT",
      "BackupRule1EnableContinuousBackup" : "true",
      "BackupRule1ScheduleExpression" : "cron(0 1 ? * * *)"
    },
    "StackTemplateId": "stm-sc68a620000000000",
    "TimeoutInMinutes": 60,
    "Name": "TEST_STACK"
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateBackupPlanRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateBackupPlanRfc.json
```

4. Modify and save the CreateBackupPlanRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId" : "ct-2hyozbpa0sx0m",
  "Version" : "2.0",
   "Title": "AWS Backup create backup plan"
}
```

5. Create the RFC, specifying the CreateBackupPlanRfc file and the CreateBackupPlanParams file:

```
aws amscm create-rfc --cli-input-json file://CreateBackupPlanRfc.json --execution-
parameters file://CreateBackupPlanParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- The continuous backup configuration requires some additional knowledge. With continuous backups, you can restore your AWS Backup-supported resource by rewinding it back to a specific time that you choose, within 1 second of precision (going back a maximum of 35 days). In AMS Advanced, you configure this with the **BackupRule1EnableContinuousBackup** parameter set to **true**, which causes AWS Backup to create continuous backups capable of point-in-time restore (PITR). To learn more, see Restoring to a specified time using Point-In-Time Recovery (PITR).
- Not all resource types supported by AWS Backup are enabled by default. Review the enabled resource types in your account using Getting Started 1: Service Opt-In.
- If you need to change your backup plan, open an RFC using the Other | Create (Review Required) CT (ct-1e1xtak34nx76).
- To learn more about AWS Backup, see AWS Backup: How It Works.
- Before creating backup plans, confirm supported resources at Feature availability by resource.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2hyozbpa0sx0m.

Example: Required Parameters

```
{
  "Description": "This is a test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "Test Stack",
  "Parameters": {
    "BackupPlanName": "MyCustomBackupPlan",
    "ResourceTagKey": "custom_backup_test",
    "ResourceTagValue": "true",
    "BackupRule1Name": "BackupRule1",
    "BackupRule1Vault": "ams-custom-backups",
    "BackupRule1CompletionWindowMinutes": 1440,
    "BackupRule1ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule1DeleteAfterDays": 0,
    "BackupRule1MoveToColdStorageAfterDays": 0,
    "BackupRule1StartWindowMinutes": 180,
    "BackupRule1RecoveryPointTagKey": "test",
    "BackupRule1RecoveryPointTagValue": "test"
  },
```

```
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-sc68a620000000000"
}
```

Example: All Parameters

```
{
  "Description": "This is a test description.",
  "VpcId": "vpc-1234567890abcdef0",
  "Name": "Test Stack",
  "Parameters": {
    "BackupPlanName": "MyCustomBackupPlan",
    "ResourceTagKey": "custom_backup_test",
    "ResourceTagValue": "true",
    "WindowsVSS" : "disabled",

    "BackupRule1Name": "BackupRule1",
    "BackupRule1Vault": "ams-custom-backups",
    "BackupRule1CompletionWindowMinutes": 1440,
    "BackupRule1ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule1DeleteAfterDays": 0,
    "BackupRule1MoveToColdStorageAfterDays": 0,
    "BackupRule1StartWindowMinutes": 180,
    "BackupRule1RecoveryPointTagKey": "test",
    "BackupRule1RecoveryPointTagValue": "test",
    "BackupRule1EnableContinuousBackup": "false",

    "BackupRule2Name": "BackupRule2",
    "BackupRule2Vault": "ams-custom-backups",
    "BackupRule2CompletionWindowMinutes": 1440,
    "BackupRule2ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule2DeleteAfterDays": 0,
    "BackupRule2MoveToColdStorageAfterDays": 0,
    "BackupRule2StartWindowMinutes": 180,
    "BackupRule2RecoveryPointTagKey": "test",
    "BackupRule2RecoveryPointTagValue": "test",
    "BackupRule2EnableContinuousBackup": "false",

    "BackupRule3Name": "BackupRule3",
    "BackupRule3Vault": "ams-custom-backups",
    "BackupRule3CompletionWindowMinutes": 1440,
    "BackupRule3ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule3DeleteAfterDays": 0,
```

```
    "BackupRule3MoveToColdStorageAfterDays": 0,
    "BackupRule3StartWindowMinutes": 180,
    "BackupRule3RecoveryPointTagKey": "test",
    "BackupRule3RecoveryPointTagValue": "test",
    "BackupRule3EnableContinuousBackup": "false",

    "BackupRule4Name": "BackupRule4",
    "BackupRule4Vault": "ams-custom-backups",
    "BackupRule4CompletionWindowMinutes": 1440,
    "BackupRule4ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule4DeleteAfterDays": 0,
    "BackupRule4MoveToColdStorageAfterDays": 0,
    "BackupRule4StartWindowMinutes": 180,
    "BackupRule4RecoveryPointTagKey": "test",
    "BackupRule4RecoveryPointTagValue": "test",
    "BackupRule4EnableContinuousBackup": "false",

    "BackupRule5Name": "BackupRule5",
    "BackupRule5Vault": "ams-custom-backups",
    "BackupRule5CompletionWindowMinutes": 1440,
    "BackupRule5ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule5DeleteAfterDays": 0,
    "BackupRule5MoveToColdStorageAfterDays": 0,
    "BackupRule5StartWindowMinutes": 180,
    "BackupRule5RecoveryPointTagKey": "test",
    "BackupRule5RecoveryPointTagValue": "test",
    "BackupRule5EnableContinuousBackup": "false",

    "BackupRule6Name": "BackupRule6",
    "BackupRule6Vault": "ams-custom-backups",
    "BackupRule6CompletionWindowMinutes": 1440,
    "BackupRule6ScheduleExpression": "cron(0 2 ? * * *)",
    "BackupRule6DeleteAfterDays": 0,
    "BackupRule6MoveToColdStorageAfterDays": 0,
    "BackupRule6StartWindowMinutes": 180,
    "BackupRule6RecoveryPointTagKey": "test",
    "BackupRule6RecoveryPointTagValue": "test",
    "BackupRule6EnableContinuousBackup": "false"

  },
  "TimeoutInMinutes": 60,
  "StackTemplateId": "stm-sc68a620000000000"
}
```

Directory Service Subcategory

Change Type Items and Operations in the Directory Service Subcategory

DNS | Create Conditional Forwarder

Create AD DNS conditional forwarder with up to five DNS servers associated with a remote domain name. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Deployment | Directory Service | DNS | Create conditional forwarder

Change Type Details

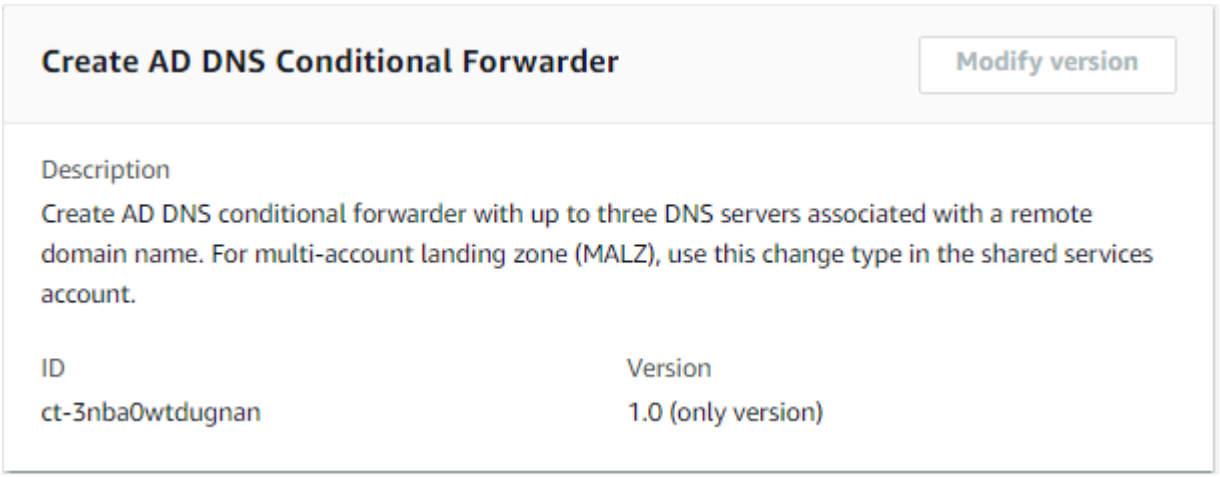
Change type ID	ct-3nba0wtdugnan
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create DNS conditional forwards

Creating DNS conditional forwards with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating DNS conditional forwards with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3nba0wtdugnan" --change-type-version "1.0"
 --title "Create conditional forwarders" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-CreateADDNSConditionalForwarder-Admin\",\"Region\": \"us-
east-1\",\"Parameters\": {\"RemoteDomainName\": [\"Domain_Name\"], \"IPAddresses\":
 [\"132.133.134.135\", \"135.134.133.132\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CondForwardCreateParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3nba0wtdugnan"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CondForwardCreateParams.json
   ```

   Modify and save the CondForwardCreateParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateADDNSConditionalForwarder-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RemoteDomainName": [
         "Domain_Name"
       ],
       "IPAddresses": [
         "132.133.134.135", "135.134.133.132"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it CondForwardCreateRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CondForwardCreateRfc.json
   ```

3. Modify and save the CondForwardCreateRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-3nba0wtdugnan",
   ```

```
"ChangeTypeVersion":    "1.0",
"Title":                "Create conditional forwarders"
}
```

4. Create the RFC, specifying the CondForwardCreateRfc file and the CondForwardCreateParams file:

```
aws amscm create-rfc --cli-input-json file://CondForwardCreateRfc.json --execution-
parameters file://CondForwardCreateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3nba0wtdugnan.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"],
    "IPAddresses": ["10.0.0.1", "10.0.0.2"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"],
    "IPAddresses": ["10.0.0.1", "10.0.0.2"]
  }
}
```

DNS | Create Group Managed Service Account

Create a new Active Directory (AD) Group Managed Service Account (gMSA). For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Deployment | Directory Service | DNS | Create group managed service account

Change Type Details

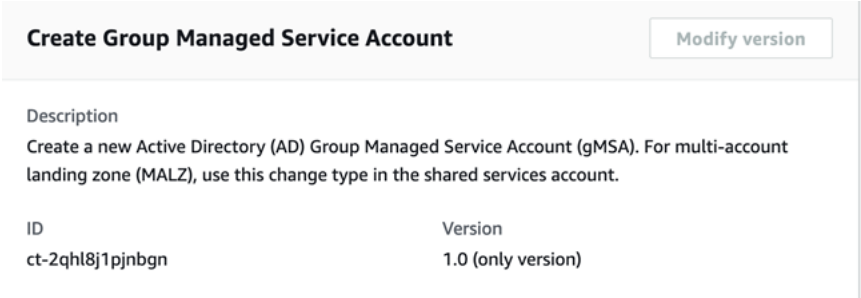
Change type ID	ct-2qhl8j1pjnbgn
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create group managed service account

Creating a group managed service account with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a group managed service account with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2qhl8j1pjnbgn" --change-type-version
 "1.0" --title "Create Group Managed Service Account" --execution-parameters
 "{\"DocumentName\": \"AWSManagedServices-CreateADGroupManagedServiceAccount-Admin
\",\"Region\": \"us-east-1\",\"Parameters\": {\"AccountName\": [\"Test-Sample\"],
\"ManagedPasswordIntervalInDays\": [\"30\"],\"PrincipalAllowedToRetrievePassword
\": [\"Test-admin\"],\"ComputerName\": [\"TestComputer\"],\"DNSHostName\":
 [\"test.domain.com\"],\"KerberosEncryptionType\": [\"RC4\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it GroupManServAcctCreateParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2qhl8j1pjnbgn"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    GroupManServAcctCreateParams.json
   ```

Modify and save the GroupManServAcctCreateParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-CreateADGroupManagedServiceAccount-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "AccountName": [
      "Test-Sample"
    ],
    "ManagedPasswordIntervalInDays": [
      "30"
    ],
    "PrincipalAllowedToRetrievePassword": [
      "Test-admin"
    ],
    "ComputerName": [
      "Test-Computer"
    ],
    "DNSHostName": [
      "test.domain.com"
    ],
    "KerberosEncryptionType": [
      "RC4"
    ]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it GroupManServAcctCreateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > GroupManServAcctCreateRfc.json
```

3. Modify and save the GroupManServAcctCreateRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2qhl8j1pjnbgn",
  "Title": "Create Group Managed Service Account"
}}
```

4. Create the RFC, specifying the GroupManServAcctCreateRfc file and the GroupManServAcctCreateParams file:

```
aws amscm create-rfc --cli-input-json file://GroupManServAcctCreateRfc.json --
execution-parameters file://GroupManServAcctCreateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2qhl8j1pjnbgn.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateADGroupManagedServiceAccount-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "AccountName": ["Sample-account"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateADGroupManagedServiceAccount-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "AccountName": ["Sample-account"],
    "ManagedPasswordIntervalInDays": ["30"],
    "PrincipalAllowedToRetrievePassword": ["Sample_Principal"],
    "ComputerName": ["Sample-Computer"],
    "DNSHostName": ["test.domain.com"],
    "KerberosEncryptionType": ["RC4,AES128,AES256"]
  }
}
```

Ingestion Subcategory

Change Type Items and Operations in the Ingestion Subcategory

- Stack from CloudFormation Template | Create
- Stack from Migration Partner Migrated Instance | Create

Stack from CloudFormation Template | Create

Create a stack by pointing to a customized CloudFormation (CFN) template in an S3 bucket, or by pasting the contents of that template as input to this change type.

Full classification: Deployment | Ingestion | Stack from CloudFormation Template | Create

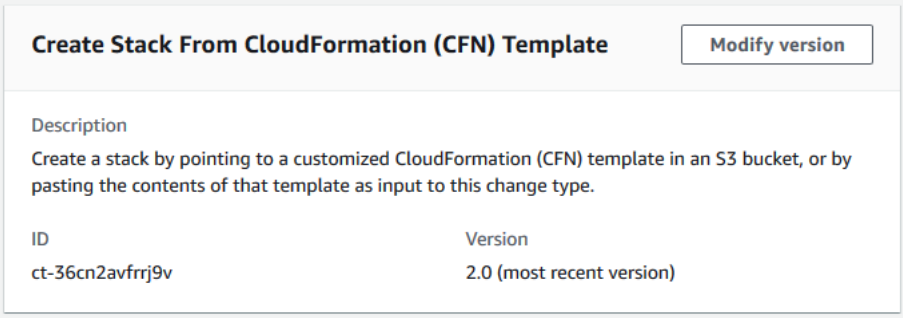
Change Type Details

Change type ID	ct-36cn2avfrrj9v
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CloudFormation ingest stack

Creating a CloudFormation ingest stack using the console



To create a CloudFormation ingest stack using the console

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CloudFormation ingest stack using the CLI

To create a CloudFormation ingest stack using the CLI

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

1. Prepare the CloudFormation template that you will use to create the stack, and upload it to your S3 bucket. For important details, see AWS CloudFormation Ingest Guidelines, Best Practices, and Limitations.
2. Create and submit the RFC to AMS:

- Create and save the execution parameters JSON file, include the CloudFormation template parameters that you want. The following example names it CreateCfnParams.json.

  Example Web application stack CreateCfnParams.json file:

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "$S3_URL",
  "TimeoutInMinutes": 120,
  "Tags": [
   {
    "Key":   "Enviroment Type"
    "Value": "Dev",
   },
   {
    "Key":   "Application"
    "Value": "PCS",
   }
  ],
  "Parameters": [
   {
    "Name": "Parameter-for-S3Bucket-Name",
    "Value":  "BUCKET-NAME"
   },
   {
    "Name": "Parameter-for-Image-Id",
    "Value":  "AMI-ID"
   }
  ],
}
```

  Example SNS topic CreateCfnParams.json file:

```
{
  "Name": "cfn-ingest",
  "Description": "CFNIngest Web Application Stack",
  "CloudFormationTemplateS3Endpoint": "$S3_URL",
  "Tags": [
    {"Key": "Enviroment Type", "Value": "Dev"}
  ],
```

```
  "Parameters": [
    {"Name": "TopicName", "Value": "MyTopic1"}
  ]
}
```

3. Create and save the RFC parameters JSON file with the following content. The following example names it CreateCfnRfc.json file:

```
{
   "ChangeTypeId": "ct-36cn2avfrrj9v",
   "ChangeTypeVersion": "2.0",
   "Title": "cfn-ingest"
}
```

4. Create the RFC, specifying the CreateCfnRfc file and the CreateCfnParams file:

```
aws amscm create-rfc --cli-input-json file://CreateCfnRfc.json  --execution-
parameters file://CreateCfnParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Note

This change type is at version 2.0 and is automated (not manually executed). This allows the CT execution to go more quickly, and, a new parameter, **CloudFormationTemplate**, allows you to paste into the RFC a custom CloudFormation template. Additionally, In this version, we do not attach the default AMS security groups if the you specify your own security groups. If you do not specify your own security groups in the request, AMS will attach the AMS default security groups. In CFN Ingest v1.0, we always appended the AMS default security groups whether or not you provided your own security groups.
AMS has enabled 17 AMS Self-Provisioned services for use in this change type. For information about supported resources, see CloudFormation Ingest Stack: Supported Resources.

Note

Version 2.0 accepts an S3 endpoint that is not a presigned URL.
If you use the previous version of this CT, the **CloudFormationTemplateS3Endpoint** parameter value must be a presigned URL.
Example command for generating a presigned S3 bucket URL (Mac/Linux):

```
export S3_PRESIGNED_URL=$(aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json)
```

Example command for generating a presigned S3 bucket URL (Windows):

```
for /f %i in ('aws s3 presign DASHDASHexpires-in 86400
 s3://BUCKET_NAME/CFN_TEMPLATE.json') do set S3_PRESIGNED_URL=%i
```

See also Creating Pre-Signed URLs for Amazon S3 Buckets.

Note

If the S3 bucket exists in an AMS account, you must use your AMS credentials for this command. For example, you may need to append `--profile saml` after obtaining your AMS AWS Security Token Service (AWS STS) credentials.

Related change types: Approve a CloudFormation ingest stack changeset, Update AWS CloudFormation ingest stack

To learn more about AWS CloudFormation, see AWS CloudFormation. To see CloudFormation templates, open the AWS CloudFormation Template Reference.

Validating a AWS CloudFormation ingest

The template is validated to ensure that it can be created in an AMS account. If it passes validation, it's updated to include any resources or configurations required for it to conform with AMS. This includes adding resources such as Amazon CloudWatch alarms in order to allow AMS Operations to monitor the stack.

The RFC is rejected if any of the following are true:

- RFC JSON Syntax is incorrect or does not follow the given format.
- The provided S3 bucket presigned URL is not valid.
- The template is not valid AWS CloudFormation syntax.
- The template does not have defaults set for all parameter values.
- The template fails AMS validation. For AMS validation steps, see the information later in this topic.

The RFC fails if the CloudFormation stack fails to create due to a resource creation issue.

To learn more about CFN validation and validator, see Template Validation and CloudFormation ingest stack: CFN validator examples.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-36cn2avfrrj9v.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Stack from Migration Partner Migrated Instance | Create

Migrate a running non-AMS instance into an AMS stack, in a given AMS-managed VPC and subnet. Must be an instance that was configured through a cloud migration service. Tags that exist on the instance to be migrated will be applied to the resources created in addition to tags requested in the RFC. Number of total tags between the instance to be migrated and the resources created cannot exceed fifty. Set a Name tag to give the EC2 instance, and AMI, names in the EC2 console. Please note that your RFC will be rejected if a tag on the instance to be migrated has the same key as a tag supplied in the RFC.

Full classification: Deployment | Ingestion | Stack from migration partner migrated instance | Create

Change Type Details

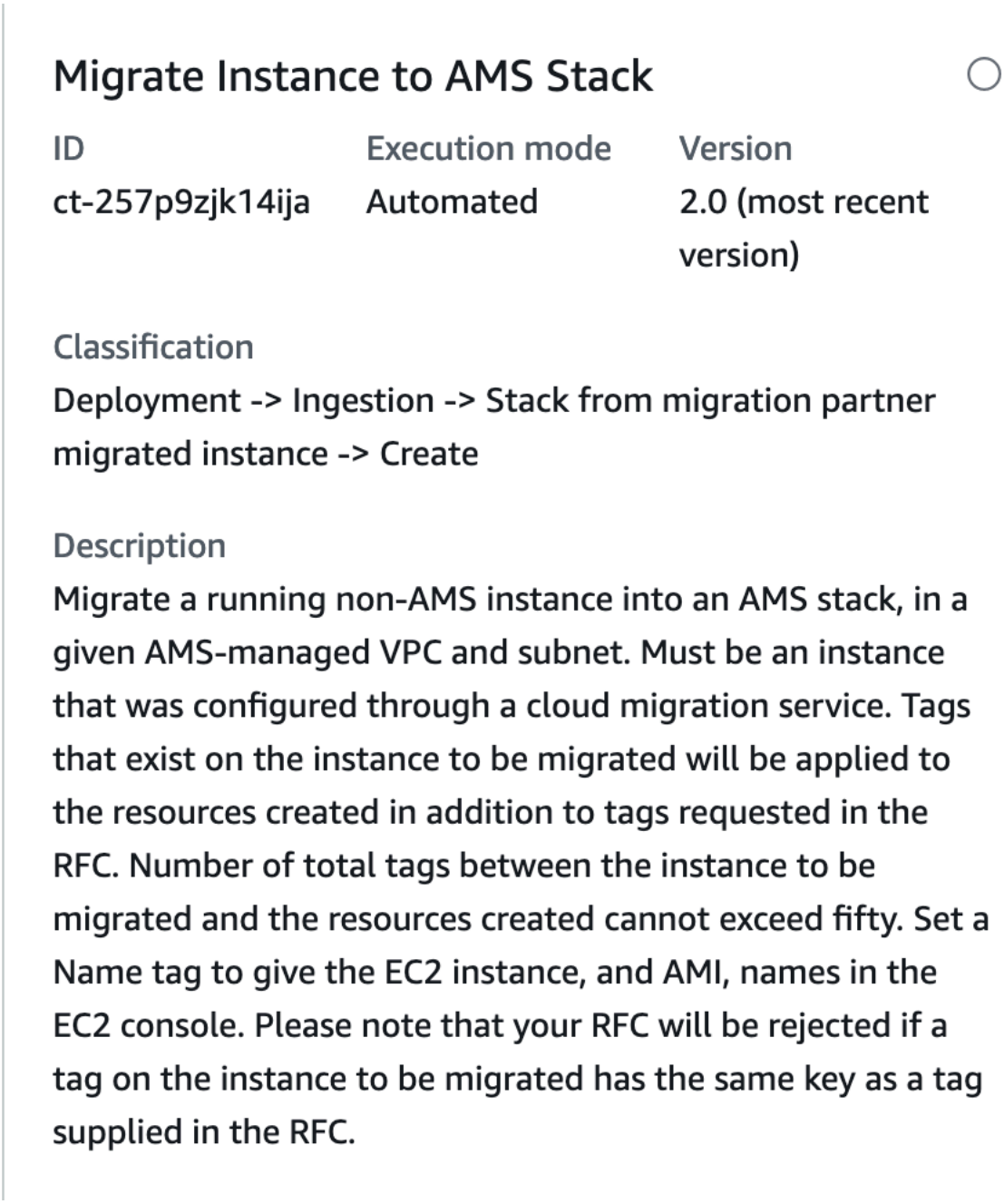
Change type ID	ct-257p9zjk14ija
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Workload Ingest Stack: Creating

Migrating an instance to an AMS stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

> **Note**
>
> If the RFC is rejected, the execution output includes a link to Amazon CloudWatch logs. AMS Workload Ingest (WIGS) RFCs are rejected when requirements are not met; for example, if anti-virus software is detected on the instance. The CloudWatch logs will include information about the failed requirement and the actions to take for remediation.

Migrating an instance to an AMS stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC

parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

You can use the AMS CLI to create an AMS instance from a non-AMS instance migrated to an AMS account.

Note

Be sure you have followed the prerequisites; see Migrating Workloads: Prerequisites for Linux and Windows.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-257p9zjk14ija" --change-type-version "2.0" --
title "AMS-WIG-TEST-NO-ACTION" --execution-parameters "{\"InstanceId\":\"INSTANCE_ID\",
\"TargetVpcId\":\"VPC_ID\",\"TargetSubnetId\":\"SUBNET_ID\",\"TargetInstanceType\":
\"t2.large\",\"ApplyInstanceValidation\":true,\"Name\":\"WIG-TEST\",\"Description\":
\"WIG-TEST\",\"EnforceIMDSV2\":\"false\"}"
```

TEMPLATE CREATE:

1. 0utput the execution parameters JSON schema for this change type to a file; example names it MigrateStackParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-257p9zjk14ija" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > MigrateStackParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "InstanceId":          "MIGRATED_INSTANCE_ID",
   "TargetVpcId":         "VPC_ID",
   "TargetSubnetId":      "SUBNET_ID",
   "Name":                "Migrated-Stack",
   "Description":         "Create-Migrated-Stack",
   "EnforceIMDSV2":       "false"
   }
   ```

3. Output the RFC template JSON file; example names it MigrateStackRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > MigrateStackRfc.json
   ```

4. Modify and save the MigrateStackRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-257p9zjk14ija",
   "ChangeTypeVersion":    "2.0",
   "Title":                "Migrate-Stack-RFC"
   ```

```
}
```

5. Create the RFC, specifying the MigrateStackRfc file and the MigrateStackParams file:

```
aws amscm create-rfc --cli-input-json file://MigrateStackRfc.json  --execution-
parameters file://MigrateStackParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

   The new instance appears in the Instance list for the application owner's account for the relevant VPC.

6. Once the RFC completes successfully, notify the application owner so he or she can log into the new instance and verify that the workload is operational.

> **Note**
>
> If the RFC is rejected, the execution output includes a link to Amazon CloudWatch logs. AMS Workload Ingest (WIGS) RFCs are rejected when requirements are not met; for example, if anti-virus software is detected on the instance. The CloudWatch logs will include information about the failed requirement and the actions to take for remediation.

Tips

> **Note**
>
> Be sure you have followed the prerequisites; see Migrating Workloads: Prerequisites for Linux and Windows.

> **Note**
>
> If a tag on the instance being migrated has the same key as a tag supplied in the RFC, the RFC fails.

Note

You can specify up to four Target IDs, Ports, and Availability Zones.

Note

If the RFC is rejected, the execution output includes a link to Amazon CloudWatch logs. AMS Workload Ingest (WIGS) RFCs are rejected when requirements are not met; for example, if anti-virus software is detected on the instance. The CloudWatch logs will include information about the failed requirement and the actions to take for remediation.

Note

If the RFC is rejected, the execution output includes a link to Amazon CloudWatch logs. AMS Workload Ingest (WIGS) RFCs are rejected when requirements are not met; for example, if anti-virus software is detected on the instance. The CloudWatch logs will include information about the failed requirement and the actions to take for remediation.

If needed, see Workload ingestion (WIGS) failure.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-257p9zjk14ija.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "InstanceId" : "i-1234567890abababa",
  "TargetVpcId" : "vpc-01234567890abcdef",
  "TargetSubnetId" : "subnet-12345678901234567",
  "TargetSecurityGroupIds": ["sg-01234567890abcdef", "sg-1234567890abcdef0"],
```

```
  "Name": "My Stack",
  "Description": "This is my stack",
  "TargetInstanceType": "t2.large",
  "ApplyInstanceValidation": true,
  "KmsKeyId": "arn:aws:kms:us-east-1:012345678910:key/a3ccc020-
abcd-1234-8d69-2f060c3c1234",
  "EnforceIMDSV2": "true",
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    },
    {
      "Key": "key3",
      "Value": "value3"
    },
    {
      "Key": "key4",
      "Value": "value4"
    },
    {
      "Key": "key5",
      "Value": "value5"
    },
    {
      "Key": "key6",
      "Value": "value6"
    },
    {
      "Key": "key7",
      "Value": "value7"
    },
    {
      "Key": "key8",
      "Value": "value8"
    },
    {
      "Key": "key9",
      "Value": "value9"
    },
```

```
    {
      "Key": "key10",
      "Value": "value10"
    },
    {
      "Key": "key11",
      "Value": "value11"
    },
    {
      "Key": "key12",
      "Value": "value12"
    },
    {
      "Key": "key13",
      "Value": "value13"
    },
    {
      "Key": "key14",
      "Value": "value14"
    },
    {
      "Key": "key15",
      "Value": "value15"
    },
    {
      "Key": "key16",
      "Value": "value16"
    },
    {
      "Key": "key17",
      "Value": "value17"
    },
    {
      "Key": "key18",
      "Value": "value18"
    },
    {
      "Key": "key19",
      "Value": "value19"
    },
    {
      "Key": "key20",
      "Value": "value20"
    },
```

```
    {
      "Key": "key21",
      "Value": "value21"
    },
    {
      "Key": "key22",
      "Value": "value22"
    },
    {
      "Key": "key23",
      "Value": "value23"
    },
    {
      "Key": "key24",
      "Value": "value24"
    },
    {
      "Key": "key25",
      "Value": "value25"
    },
    {
      "Key": "key26",
      "Value": "value26"
    },
    {
      "Key": "key27",
      "Value": "value27"
    },
    {
      "Key": "key28",
      "Value": "value28"
    },
    {
      "Key": "key29",
      "Value": "value29"
    },
    {
      "Key": "key30",
      "Value": "value30"
    },
    {
      "Key": "key31",
      "Value": "value31"
    },
```

```
    {
      "Key": "key32",
      "Value": "value32"
    },
    {
      "Key": "key33",
      "Value": "value33"
    },
    {
      "Key": "key34",
      "Value": "value34"
    },
    {
      "Key": "key35",
      "Value": "value35"
    },
    {
      "Key": "key36",
      "Value": "value36"
    },
    {
      "Key": "key37",
      "Value": "value37"
    },
    {
      "Key": "key38",
      "Value": "value38"
    },
    {
      "Key": "key39",
      "Value": "value39"
    },
    {
      "Key": "key40",
      "Value": "value40"
    },
    {
      "Key": "key41",
      "Value": "value41"
    },
    {
      "Key": "key42",
      "Value": "value42"
    },
```

```
    {
      "Key": "key43",
      "Value": "value43"
    },
    {
      "Key": "key44",
      "Value": "value44"
    },
    {
      "Key": "key45",
      "Value": "value45"
    },
    {
      "Key": "key46",
      "Value": "value46"
    },
    {
      "Key": "key47",
      "Value": "value47"
    },
    {
      "Key": "key48",
      "Value": "value48"
    },
    {
      "Key": "key49",
      "Value": "value49"
    },
    {
      "Key": "key50",
      "Value": "value50"
    }
  ]
}
```

Managed Firewall Subcategory

Change Type Items and Operations in the Managed Firewall Subcategory

- Outbound (Palo Alto) | Create Allow List
- Outbound (Palo Alto) | Create Security Policy

Outbound (Palo Alto) | Create Allow List

Create an allow list file for AMS managed Palo Alto firewall - Outbound.

Full classification: Deployment | Managed Firewall | Outbound (Palo Alto) | Create allow list

Change Type Details

Change type ID	ct-309eozh6lpkr8
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create allow list for managed Palo Alto outbound firewall

Creating an allow list for your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an allow list for your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-309eozh6lpkr8" --change-type-version "1.0" --
title "Create Allow List" --execution-parameters "{ \"RequestType\": \"CreateAllowList
\", \"Parameters\": { "AllowListName": "CustomAllowList" } } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAllowListParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-309eozh6lpkr8"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateAllowListParams.json
   ```

2. Modify and save the CreateAllowListParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "CreateAllowList",
     "Parameters": {
         "AllowListName": "CustomAllowList"
     }
   }
   ```

3. Output the RFC template JSON file to a file named CreateAllowListRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAllowListRfc.json
```

4. Modify and save the CreateAllowListRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-309eozh6lpkr8",
"Title":                "Create-Allow-List-RFC"
}
```

5. Create the RFC, specifying the CreateAllowList Rfc file and the CreateAllowListParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAllowListRfc.json  --execution-
parameters file://CreateAllowListParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-309eozh6lpkr8.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Parameters": {
    "AllowListName": "test_file"
  },
  "RequestType": "CreateAllowList"
```

```
}
```

Outbound (Palo Alto) | Create Security Policy

Create a security policy for AMS managed Palo Alto firewall - Outbound.

Full classification: Deployment | Managed Firewall | Outbound (Palo Alto) | Create security policy

Change Type Details

Change type ID	ct-281dpwh9tqnan
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create security policy for managed Palo Alto outbound firewall

Creating a security policy for your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a security policy for your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-281dpwh9tqnan" --change-type-version
 "1.0" --title "Create Security Policy" --execution-parameters "{ \"RequestType\":
 \"CreateSecurityPolicy\", \"Parameters\": { "SecurityPolicyName": "custom-sec-name",
 "SourceAddresses": ["1.0.0.0"], "DestinationAddresses": ["2.0.0.0"], "AllowLists":
 ["CustomAllowList"], "ServicePorts": { "tcp": [20]: "udp": [30] }} } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateSecurityPolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-281dpwh9tqnan"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateSecurityPolicyParams.json
   ```

2. Modify and save the CreateSecurityPolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "CreateSecurityPolicy",
     "Parameters": {
         "SecurityPolicyName": "custom-sec-name",
         "SourceAddresses": ["1.0.0.0"],
         "DestinationAddresses": ["2.0.0.0"],
   ```

```
      "AllowLists": [],
      "ServicePorts": {
            "tcp": [20]
            "udp": [30]
      }
  }
}
```

3. Output the RFC template JSON file to a file named CreateSecurityPolicyRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateSecurityPolicyRfc.json
```

4. Modify and save the CreateSecurityPolicyRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-281dpwh9tqnan",
"Title":                "Create-Security-Policy-RFC"
}
```

5. Create the RFC, specifying the CreateSecurityPolicy Rfc file and the CreateSecurityPolicyParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSecurityPolicyRfc.json  --
execution-parameters file://CreateSecurityPolicyParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This automated change type applies only to AMS multi-account landing zone (MALZ).

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-281dpwh9tqnan.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "Parameters": {
      "SecurityPolicyName": "custom-sec-pol",
      "SourceAddresses": ["10.0.0.1", "10.50.0.0/16"],
      "DestinationAddresses": ["1.1.1.1", "100.0.0.0/8", "amazon.com"],
      "ServicePorts": { "tcp": [1000, 1200] },
      "ActionType": "Allow",
      "EnablePolicy": false
    },
    "RequestType": "CreateSecurityPolicy"
}
```

Managed Landing Zone Subcategory

Change Type Items and Operations in the Managed Landing Zone Subcategory

- Application Account | Create VPC
- Application Account | Create VPC Additional CIDR and Subnets
- Management Account | Create Accelerate Account
- Management Account | Create Application Account (With VPC)
- Management Account | Create Custom OUs
- Management Account | Create Custom SCP (Review Required)
- Management Account | Create Customer-Managed Application Account
- Management Account | Create Developer Mode Account (With VPC)
- Management Account | Create StackSets Stack (Review Required)
- Management Account | Create Tools Account (With VPC)
- Networking Account | Add Static Route
- Networking Account | Create Application Route Table (Review Required)
- Networking Account | Create Transit Gateway Route Table

Application Account | Create VPC

Create a VPC with up to 10 private subnets and up to 5 optional public subnets per availability zone (AZ) for two or three AZ's.

Full classification: Deployment | Managed landing zone | Application account | Create VPC

Change Type Details

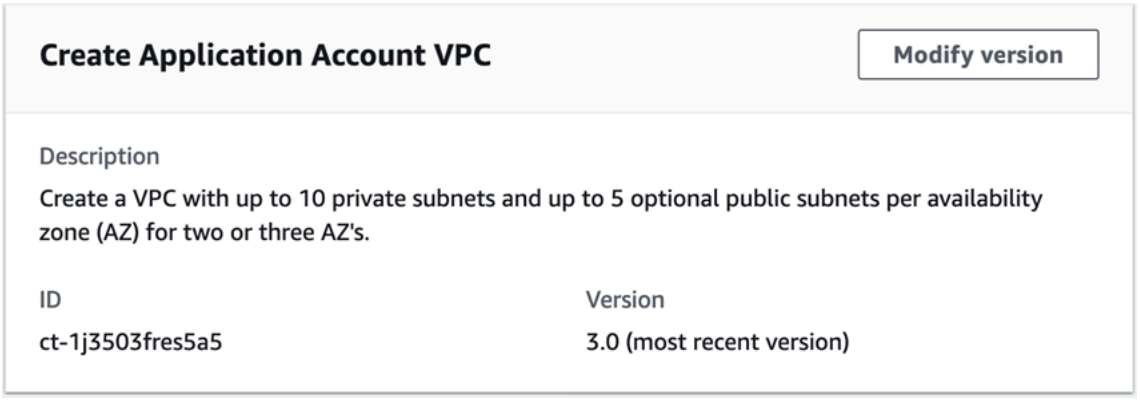
Change type ID	ct-1j3503fres5a5
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create VPC

Application account: Creating a VPC with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Application account: Creating a VPC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Application account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1j3503fres5a5" --change-type-version "3.0"
 --title "Application account VPC onboarding" --execution-parameters "{\"VpcName\":
 \"VPC_NAME\", \"Parameters\": { \"NumberOfAZs\": \"INTEGER\", \"VPCCIDR\": \"X.X.X.X/
X\", \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",\"PrivateSubnet1AZ2CIDR\": \"X.X.X.X/
X\", \"RouteType\": \"ROUTE_TYPE\", \"TransitGatewayApplicationRouteTableName\":
 \"TABLE_NAME\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAppAcctVpcParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1j3503fres5a5"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateAppAcctVpcParams.json
```

2. Modify and save the CreateAppAcctVpcParams file. For example, you can replace the contents with something like this:

```
{
"VpcName": "TestVPC",
"Parameters": {
  "NumberOfAZs": "INTEGER",
  "VPCCIDR": "x.x.x.x/x",
  "PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
  "PrivateSubnet1AZ2CIDR": "x.x.x.x/x",
  "PrivateSubnet1AZ3CIDR": "x.x.x.x/x",
  "PublicSubnetAZ1CIDR": "x.x.x.x/x",
  "PublicSubnetAZ2CIDR": "x.x.x.x/x",
  "PublicSubnetAZ3CIDR": "x.x.x.x/x",
  "RouteType": "ROUTE_TYPE",
  "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME"
}
}
```

3. Output the RFC template JSON file to a file; this example names it CreateAppAcctVpcRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAppAcctVpcRfc.json
```

4. Modify and save the CreateAppAcctVpcRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-1j3503fres5a5",
"Title":                "App-Acct-Vpc-RFC"
}
```

5. Create the RFC, specifying the CreateAppAcctVpcRfc file and the CreateAppAcctVpcParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAppAcctVpcRfc.json  --execution-
parameters file://CreateAppAcctVpcParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- **Important**

  To create an additional public subnet in a new availability zone (AZ), a private subnet must already be present.

- This change type is now at version 3.0 and it has been automated (it is no longer manually run by AMS). The 2.0 version of this change type was a "review required" (manual) change type.
- To learn more about AMS multi-account landing zone, see VPC sharing: A new approach to multiple accounts and VPC management.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1j3503fres5a5.

Example: Required Parameters

```
{
  "VpcName": "TestVPC",
  "Parameters": {
    "VPCCIDR": "10.0.0.0/22",
    "NumberOfAZs": 2,
    "PrivateSubnet1AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet1AZ2CIDR": "10.0.1.0/24"
  }
}
```

Example: All Parameters

```
{
```

```
  "VpcName": "TestVPC",
  "Parameters": {
    "VPCCIDR": "10.0.0.0/22",
    "NumberOfAZs": 3,
    "RouteType": "isolated",
    "TransitGatewayApplicationRouteTableName": "applications",
    "PublicSubnetAZ1CIDR": "10.0.0.0/24",
    "PublicSubnetAZ2CIDR": "10.0.1.0/24",
    "PublicSubnetAZ3CIDR": "10.0.2.0/24",
    "PublicSubnet2AZ1CIDR": "10.0.0.0/24",
    "PublicSubnet2AZ2CIDR": "10.0.1.0/24",
    "PublicSubnet2AZ3CIDR": "10.0.2.0/24",
    "PublicSubnet3AZ1CIDR": "10.0.0.0/24",
    "PublicSubnet3AZ2CIDR": "10.0.1.0/24",
    "PublicSubnet3AZ3CIDR": "10.0.2.0/24",
    "PublicSubnet4AZ1CIDR": "10.0.0.0/24",
    "PublicSubnet4AZ2CIDR": "10.0.1.0/24",
    "PublicSubnet4AZ3CIDR": "10.0.2.0/24",
    "PublicSubnet5AZ1CIDR": "10.0.0.0/24",
    "PublicSubnet5AZ2CIDR": "10.0.1.0/24",
    "PublicSubnet5AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet1AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet1AZ2CIDR": "10.0.1.0/24",
    "PrivateSubnet1AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet2AZ1CIDR": "10.0.3.0/24",
    "PrivateSubnet2AZ2CIDR": "10.0.4.0/24",
    "PrivateSubnet2AZ3CIDR": "10.0.5.0/24",
    "PrivateSubnet3AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet3AZ2CIDR": "10.0.1.0/24",
    "PrivateSubnet3AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet4AZ1CIDR": "10.0.3.0/24",
    "PrivateSubnet4AZ2CIDR": "10.0.4.0/24",
    "PrivateSubnet4AZ3CIDR": "10.0.5.0/24",
    "PrivateSubnet5AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet5AZ2CIDR": "10.0.1.0/24",
    "PrivateSubnet5AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet6AZ1CIDR": "10.0.3.0/24",
    "PrivateSubnet6AZ2CIDR": "10.0.4.0/24",
    "PrivateSubnet6AZ3CIDR": "10.0.5.0/24",
    "PrivateSubnet7AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet7AZ2CIDR": "10.0.1.0/24",
    "PrivateSubnet7AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet8AZ1CIDR": "10.0.3.0/24",
    "PrivateSubnet8AZ2CIDR": "10.0.4.0/24",
```

```
    "PrivateSubnet8AZ3CIDR": "10.0.5.0/24",
    "PrivateSubnet9AZ1CIDR": "10.0.0.0/24",
    "PrivateSubnet9AZ2CIDR": "10.0.1.0/24",
    "PrivateSubnet9AZ3CIDR": "10.0.2.0/24",
    "PrivateSubnet10AZ1CIDR": "10.0.3.0/24",
    "PrivateSubnet10AZ2CIDR": "10.0.4.0/24",
    "PrivateSubnet10AZ3CIDR": "10.0.5.0/24"
  }
}
```

Application Account | Create VPC Additional CIDR and Subnets

Create an additional VPC CIDR, or subnets, or both, for an existing application account VPC. Add up to five public and twenty private subnet tiers to the additional CIDR, or to existing CIDRs under the VPC. A subnet tier is a set of subnets provisioned in two or three Availability Zones (AZ).

Full classification: Deployment | Managed landing zone | Application account | Create VPC Additional CIDR and Subnets

Change Type Details

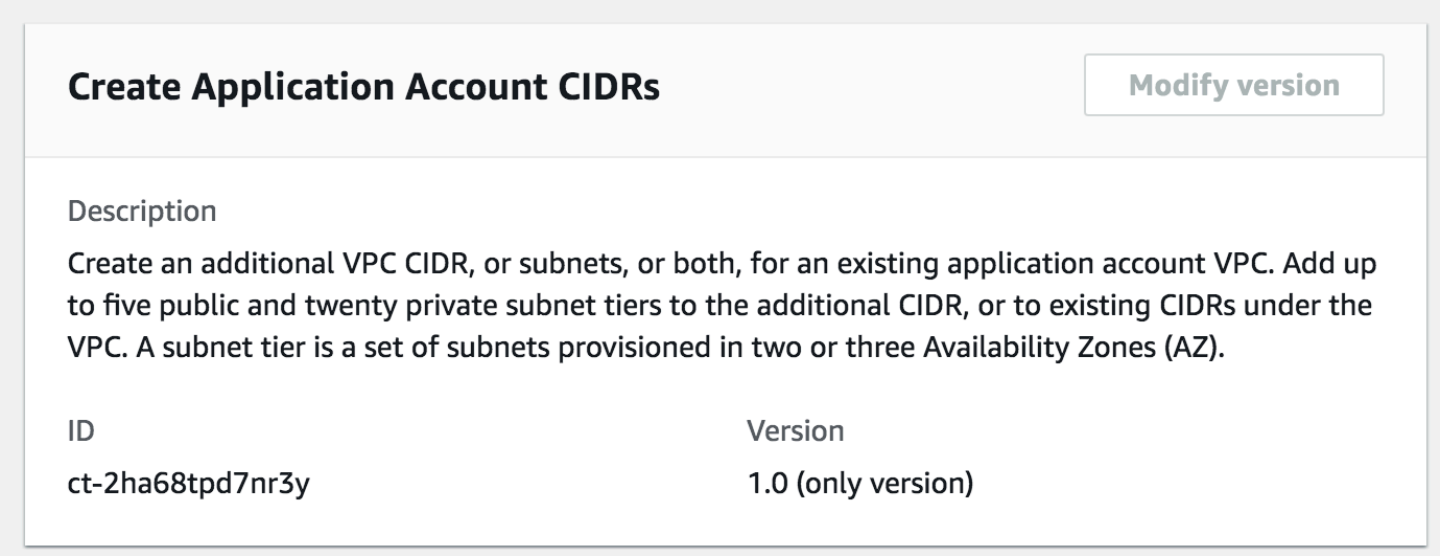
Change type ID	ct-2ha68tpd7nr3y
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create VPC CIDRs and subnets

Application account: creating VPC CIDRs or Subnets with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Application account: creating VPC CIDRs or Subnets with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Application account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

To create only additional VPC CIDRs:

```
aws amscm create-rfc --change-type-id "ct-2ha68tpd7nr3y" --change-type-version "1.0"
 --title "Additional VPC CIDR Creation" --execution-parameters "{\"VPCId\": \"VPC_ID\",
 \"Parameters\": { \"VPCCIDR\": \"X.X.X.X/X\"}}"
```

To create only additional subnets:

```
aws amscm create-rfc --change-type-id "ct-2ha68tpd7nr3y" --change-type-version
 "1.0" --title "Additional VPC Subnet Creation" --execution-parameters "{\"VPCId\":
 \"VPC_ID\", \"Parameters\": {\"PrivateRouteTableAZ1ID\": \"Transit Gateway Route Table
 AZ1 Name\", \"PrivateRouteTableAZ2ID\": \"Transit Gateway Route Table AZ2 Name\",
 \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",\"PrivateSubnet1AZ2CIDR\": \"X.X.X.X/X\"}}"
```

To create additional VPC CIDR and subnets:

```
aws amscm create-rfc --change-type-id "ct-2ha68tpd7nr3y" --change-type-version "1.0"
 --title "Additional VPC CIDR and subnet Creation" --execution-parameters "{\"VPCId\":
 \"VPC_ID\", \"Parameters\": { \"VPCCIDR\": \"X.X.X.X/X\", \"PrivateRouteTableAZ1ID
\": \"Transit Gateway Route Table AZ1 Name\", \"PrivateRouteTableAZ2ID\": \"Transit
 Gateway Route Table AZ2 Name\", \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",
\"PrivateSubnet1AZ2CIDR\": \"X.X.X.X/X\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAppAcctVpcCidrSubnetParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2ha68tpd7nr3y"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateAppAcctVpcCidrSubnetParams.json
   ```

2. Modify and save the CreateAppAcctVpcCidrSubnetParams file. For example, you can replace the contents with something like this:

   To create only additional VPC CIDRs:

```
{
{
"VPCId": "VPC_ID",
 "Parameters": {
   "VPCCIDR": "x.x.x.x/x",
}
}
}
```

   To create only additional subnets:

```
{
"VPCId": "VPC_ID",
"Parameters": {
"PrivateRouteTableAZ1ID": "Transit Gateway Route Table AZ1 Name",
"PrivateRouteTableAZ2ID": "Transit Gateway Route Table AZ2 Name",
"PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
"PrivateSubnet1AZ2CIDR": "x.x.x.x/x"
}
}
```

   To create additional VPC CIDR and subnets:

```
{
"VPCId": "VPC_ID",
"Parameters": {
"VPCCIDR": "x.x.x.x/x",
"PrivateRouteTableAZ1ID": "Transit Gateway Route Table AZ1 Name",
"PrivateRouteTableAZ2ID": "Transit Gateway Route Table AZ2 Name",
"PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
"PrivateSubnet1AZ2CIDR": "x.x.x.x/x"
}
}
```

3. Output the RFC template JSON file to a file; this example names it CreateAppAcctVpcCidrSubnetRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAppAcctVpcCidrSubnetRfc.json
```

4. Modify and save the CreateAppAcctVpcCidrSubnetRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2ha68tpd7nr3y",
"Title":                "App-Acct-Vpc-Cidr-Subnets-RFC"
}
```

5. Create the RFC, specifying the CreateAppAcctVpcCidrSubnetRfc file and the CreateAppAcctVpcCidrSubnetParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAppAcctVpcCidrSubnetRfc.json  --
execution-parameters file://CreateAppAcctVpcCidrSubnetParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- **Important**

  To create an additional public subnet in a new availability zone (AZ), a private subnet must already be present.

- To use this CT to create additional public subnets in an already-provisioned VPC, the VPC must already have public subnets inside it. If this is not the case, contact AMS to deploy those public subnets inside the VPC first.
- To learn more about AMS multi-account landing zone, see VPC sharing: A new approach to multiple accounts and VPC management.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2ha68tpd7nr3y.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Management Account | Create Accelerate Account

Create an Accelerate account in your AMS-managed landing zone. Accelerate provides patching, backup, monitoring and reports, but no requests for change.

Full classification: Deployment | Managed landing zone | Management account | Create Accelerate account

Change Type Details

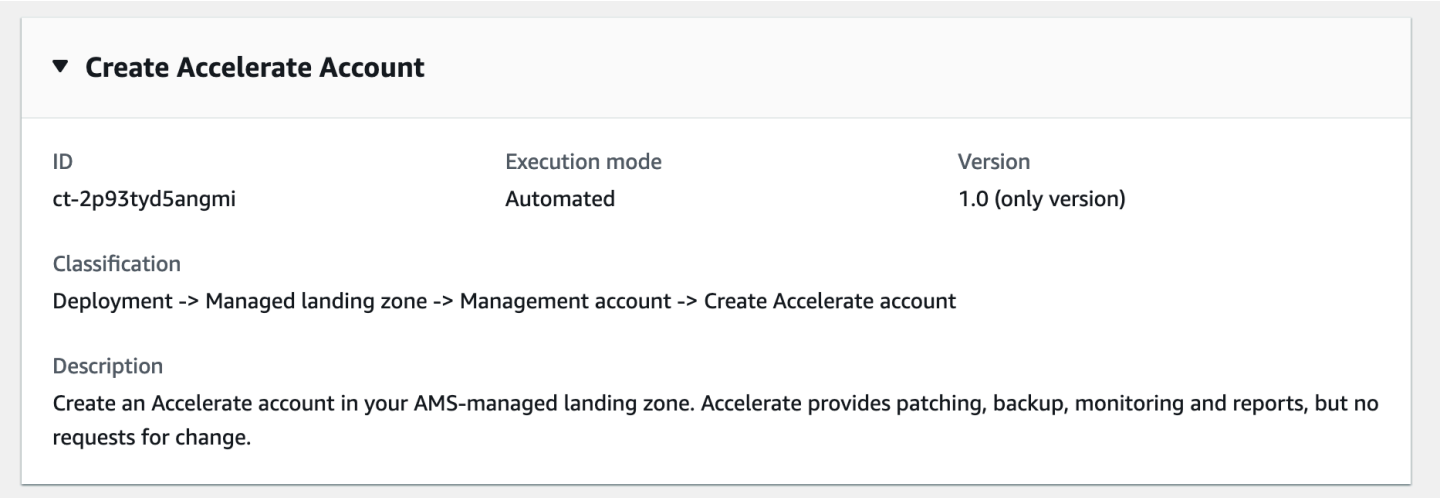
Change type ID	ct-2p93tyd5angmi
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create an Accelerate account

Management account: Creating an Accelerate account with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating an Accelerate account with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2p93tyd5angmi" --change-type-version "1.0" --
title "Create Accelerate account" --execution-parameters "{\"AccountName\": \"account-
name-1\",\"Regions\": [\"us-east-1\", \"us-east-2\"],\"AccountEmail\": \"account-
email-1@example.com\", \"AccelerateOUName\":\"accelerate\", \"SupportLevel\":\"plus\",
 \"EnablePatch\":true}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAccAcctParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1zdasmc2ewzrs" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateAccAcctParams.json
   ```

2. Modify and save the CreateAccAcctParams file. For example, you can replace the contents with something like this:

   ```
   {
    "AccountName": "AccountName",
    "AccountEmail": "nobody@amazon.com",
    "AccelerateOUName": "accelerate",
    "Regions": [
    "ap-northeast-1",
    "ap-northeast-2"
    ],
    "SupportLevel": "plus",
    "EnablePatch": true
   }
   ```

3. Output the RFC template JSON file to a file; this example names it CreateAccAcctRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateAccAcctRfc.json
   ```

4. Modify and save the CreateAccAcctRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2p93tyd5angmi",
"Title":                "Create-Accelerate-Acct"
}
```

5. Create the RFC, specifying the CreateAccAcct Rfc file and the CreateAccAcctParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAccAcctRfc.json  --execution-
parameters file://CreateAccAcctParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AMS Accelerate, see What is AMS Accelerate?.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2p93tyd5angmi.

Example: Required Parameters

```
{
  "AccountName": "AccountName",
  "AccountEmail": "nobody@amazon.com",
  "AccelerateOUName": "accelerate",
  "Regions": [
    "ap-northeast-1",
    "ap-northeast-2"
  ],
  "SupportLevel": "plus",
  "EnablePatch": true
}
```

Example: All Parameters

```
{
  "AccountName": "AccountName",
  "AccountEmail": "nobody@amazon.com",
```

```
  "AccelerateOUName": "accelerate",
  "Regions": [
    "ap-northeast-1",
    "ap-northeast-2"
  ],
  "SupportLevel": "plus",
  "EnablePatch": true
}
```

Management Account | Create Application Account (With VPC)

Create a managed AWS landing zone application account and a VPC with up to 10 private subnets and up to 5 optional public subnets per availability zone (AZ) for two or three AZ's. Optionally, also create an AWS Backup plan with up to four different rules. Managed AWS landing zone core accounts must already be onboarded to AWS Managed Services (AMS).

Full classification: Deployment | Managed landing zone | Management account | Create application account (with VPC)

Change Type Details

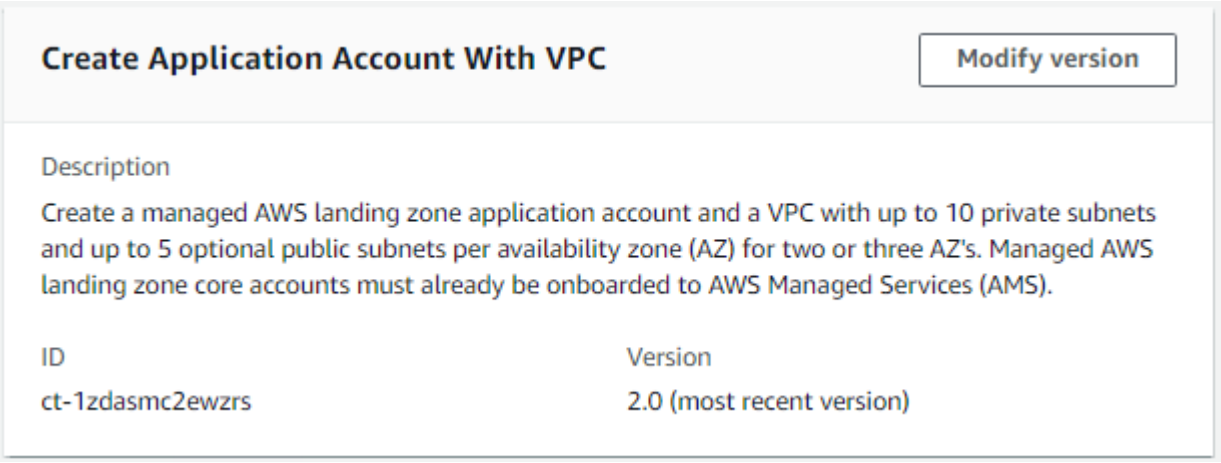
Change type ID	ct-1zdasmc2ewzrs
Current version	2.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create an Application account with VPC

Management account: Creating an application account (with VPC) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating an application account (with VPC) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1zdasmc2ewzrs" --change-type-version "2.0"
    --title "Application account onboarding" --execution-parameters "{\"AccountName
    \": \"ACCOUNT_NAME\",\"AccountEmail\": \"EMAIL_ADDRESS\",\"ApplicationOUName\":
    \"APP_ACCOUNT_OU_NAME:CHILD_OU_NAME\",\"SupportLevel\": \"LEVEL\",\"VpcName\":
    \"VPC_NAME\",\"NumberOfAZs\": \"INTEGER\",\"VpcCIDR\":
    \"X.X.X.X/X\", \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",\"PrivateSubnet1AZ2CIDR\":
    \"X.X.X.X/X\",\"PrivateSubnet1AZ3CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ1CIDR\":
    \"X.X.X.X/X\",\"PublicSubnetAZ2CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ3CIDR\":
    \"X.X.X.X/X\", \"RouteType\": \"ROUTE_TYPE\",
 \"TransitGatewayApplicationRouteTableName\":
    \"TABLE_NAME\"}"
```

With backup parameters:

```
aws amscm create-rfc --change-type-id "ct-1zdasmc2ewzrs" --change-type-version "2.0"
    --title "Application account onboarding" --execution-parameters "{\"AccountName
    \": \"ACCOUNT_NAME\",\"AccountEmail\": \"EMAIL_ADDRESS\",\"ApplicationOUName\":
    \"APP_ACCOUNT_OU_NAME:CHILD_OU_NAME\",\"SupportLevel\": \"LEVEL\",\"VpcName\":
    \"VPC_NAME\",\"NumberOfAZs\": \"INTEGER\",\"VpcCIDR\":
    \"X.X.X.X/X\", \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",\"PrivateSubnet1AZ2CIDR\":
    \"X.X.X.X/X\",\"PrivateSubnet1AZ3CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ1CIDR\":
    \"X.X.X.X/X\",\"PublicSubnetAZ2CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ3CIDR\":
    \"X.X.X.X/X\", \"RouteType\": \"ROUTE_TYPE\",
 \"TransitGatewayApplicationRouteTableName\":
    \"TABLE_NAME\", \"BackupPlanName\":\"PLAN_NAME\", \"ResourceTagKey\":
    \"TAG_KEY\", \"ResourceTagValue\":\"TAG_VALUE\", "\BackupRule1ScheduleExpression\":
    \"cron(0 2 ? * * *)\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateMgmtAcctAppAcctWithVpcParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1zdasmc2ewzrs"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateMgmtAcctAppAcctWithVpcParams.json
   ```

2. Modify and save the CreateMgmtAcctAppAcctWithVpcParams file. For example, you can replace the contents with something like this:

```
{
      "AccountName": "ACCOUNT_NAME",
      "AccountEmail": "ACCOUNT_EMAIL",
      "ApplicationOUName": "APPLICATION_OU_NAME:CHILD_OU_NAME",
      "SupportLevel": "PLUS_or_PREMIUM",
      "VpcName": "VPC_NAME",
      "NumberOfAZs": "TWO_or_THREE",
      "VpcCIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ2CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ3CIDR": "x.x.x.x/x",
      "PublicSubnetAZ1CIDR": "x.x.x.x/x",
      "PublicSubnetAZ2CIDR": "x.x.x.x/x",
      "PublicSubnetAZ3CIDR": "x.x.x.x/x",
      "RouteType": "ROUTABLE_or_ISOLATED",
      "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME"
}
```

With backup AND patch parameters:

```
{
      "AccountName": "ACCOUNT_NAME",
      "AccountEmail": "ACCOUNT_EMAIL",
      "ApplicationOUName": "APPLICATION_OU_NAME:CHILD_OU_NAME",
      "SupportLevel": "PLUS_or_PREMIUM",
      "VpcName": "VPC_NAME",
      "NumberOfAZs": "TWO_or_THREE",
      "VpcCIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ2CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ3CIDR": "x.x.x.x/x",
      "PublicSubnetAZ1CIDR": "x.x.x.x/x",
      "PublicSubnetAZ2CIDR": "x.x.x.x/x",
      "PublicSubnetAZ3CIDR": "x.x.x.x/x",
      "RouteType": "ROUTABLE_or_ISOLATED",
      "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME",
      "BackupPlanName": "PLAN_NAME",
      "ResourceTagKey": "TAG_KEY",
      "ResourceTagValue": "TAG_VALUE",
```

```
      "BackupRule1ScheduleExpression": "cron(0 2 ? * * *),"
      "PatchOrchestratorFirstTagKey": "TAG_KEY",
      "PatchOrchestratorDefaultMaintenanceWindowCutoff": "INTEGER",
      "PatchOrchestratorDefaultMaintenanceWindowDuration": "INTEGER",
      "PatchOrchestratorDefaultMaintenanceWindowSchedule": "cron(0 18 * * ? *),"
      "PatchOrchestratorDefaultMaintenanceWindowTimeZone": "TIME_ZONE",
      "PatchOrchestratorDefaultPatchBackupRetentionInDays": "INTEGER",
      "PatchOrchestratorNotificationEmails": "DISTRO_EMAIL"
}
```

3. Output the RFC template JSON file to a file; this example names it CreateMgmtAcctAppAcctWithVpcRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateMgmtAcctAppAcctWithVpcRfc.json
```

4. Modify and save the CreateMgmtAcctAppAcctWithVpcRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-1zdasmc2ewzrs",
"Title":                "Management-Acct-App-Acct-With-Vpc-RFC"
}
```

5. Create the RFC, specifying the CreateMgmtAcctAppAcctWithVpcRfc file and the CreateMgmtAcctAppAcctWithVpcParams file:

```
aws amscm create-rfc --cli-input-json file://CreateMgmtAcctAppAcctWithVpcRfc.json
 --execution-parameters file://CreateMgmtAcctAppAcctWithVpcParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> The minimum value for this parameter has changed from 60 to 1.

Important

This change type has been automated and you can now configure the VPC to have up to 10 private subnets and up to 5 public subnets. Additionally, you can now configure backup and patching.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

How do I create a managed AWS landing zone application account with a VPC?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1zdasmc2ewzrs.

Example: Required Parameters

```
{
  "AccountName": "AccountName",
  "AccountEmail": "nobody@amazon.com",
  "SupportLevel": "plus",
  "VpcName": "TestVPC",
  "VpcCIDR": "10.0.0.0/22",
  "NumberOfAZs": 2,
  "PrivateSubnet1AZ1CIDR": "10.0.0.0/24",
  "PrivateSubnet1AZ2CIDR": "10.0.1.0/24",
  "BackupPlanName": "default-backup-plan",
  "ResourceTagKey": "Backup",
  "ResourceTagValue": "True",
  "BackupRule1ScheduleExpression": "cron(0 2 ? * * )"
}
```

Example: All Parameters

```
{
  "AccountName": "AccountName",
  "AccountEmail": "nobody@amazon.com",
  "ApplicationOUName": "applications",
  "SupportLevel": "plus",
  "VpcName": "TestVPC",
```

```
  "VpcCIDR": "10.0.0.0/22",
  "NumberOfAZs": 3,
  "RouteType": "isolated",
  "TransitGatewayApplicationRouteTableName": "defaultAppRouteTable",
  "PublicSubnetAZ1CIDR": "10.0.0.0/24",
  "PublicSubnetAZ2CIDR": "10.0.1.0/24",
  "PublicSubnetAZ3CIDR": "10.0.2.0/24",
  "PublicSubnet2AZ1CIDR": "10.0.0.0/24",
  "PublicSubnet2AZ2CIDR": "10.0.1.0/24",
  "PublicSubnet2AZ3CIDR": "10.0.2.0/24",
  "PublicSubnet3AZ1CIDR": "10.0.0.0/24",
  "PublicSubnet3AZ2CIDR": "10.0.1.0/24",
  "PublicSubnet3AZ3CIDR": "10.0.2.0/24",
  "PublicSubnet4AZ1CIDR": "10.0.0.0/24",
  "PublicSubnet4AZ2CIDR": "10.0.1.0/24",
  "PublicSubnet4AZ3CIDR": "10.0.2.0/24",
  "PublicSubnet5AZ1CIDR": "10.0.0.0/24",
  "PublicSubnet5AZ2CIDR": "10.0.1.0/24",
  "PublicSubnet5AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet1AZ1CIDR": "10.0.0.0/24",
  "PrivateSubnet1AZ2CIDR": "10.0.1.0/24",
  "PrivateSubnet1AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet2AZ1CIDR": "10.0.3.0/24",
  "PrivateSubnet2AZ2CIDR": "10.0.4.0/24",
  "PrivateSubnet2AZ3CIDR": "10.0.5.0/24",
  "PrivateSubnet3AZ1CIDR": "10.0.0.0/24",
  "PrivateSubnet3AZ2CIDR": "10.0.1.0/24",
  "PrivateSubnet3AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet4AZ1CIDR": "10.0.3.0/24",
  "PrivateSubnet4AZ2CIDR": "10.0.4.0/24",
  "PrivateSubnet4AZ3CIDR": "10.0.5.0/24",
  "PrivateSubnet5AZ1CIDR": "10.0.0.0/24",
  "PrivateSubnet5AZ2CIDR": "10.0.1.0/24",
  "PrivateSubnet5AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet6AZ1CIDR": "10.0.3.0/24",
  "PrivateSubnet6AZ2CIDR": "10.0.4.0/24",
  "PrivateSubnet6AZ3CIDR": "10.0.5.0/24",
  "PrivateSubnet7AZ1CIDR": "10.0.0.0/24",
  "PrivateSubnet7AZ2CIDR": "10.0.1.0/24",
  "PrivateSubnet7AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet8AZ1CIDR": "10.0.3.0/24",
  "PrivateSubnet8AZ2CIDR": "10.0.4.0/24",
  "PrivateSubnet8AZ3CIDR": "10.0.5.0/24",
  "PrivateSubnet9AZ1CIDR": "10.0.0.0/24",
```

```
  "PrivateSubnet9AZ2CIDR": "10.0.1.0/24",
  "PrivateSubnet9AZ3CIDR": "10.0.2.0/24",
  "PrivateSubnet10AZ1CIDR": "10.0.3.0/24",
  "PrivateSubnet10AZ2CIDR": "10.0.4.0/24",
  "PrivateSubnet10AZ3CIDR": "10.0.5.0/24",
  "DirectAlertsEmail": "test@amazon.com",
  "SamlMetadataDocumentURL": "https://test.com",
  "BackupPlanName": "default-backup-plan",
  "ResourceTagKey": "Backup",
  "ResourceTagValue": "True",
  "BackupRule1ScheduleExpression": "cron(0 2 ? * * )",
  "PatchOrchestratorFirstTagKey": "AppId",
  "PatchOrchestratorSecondTagKey": "Environment",
  "PatchOrchestratorDefaultMaintenanceWindowCutoff": 1,
  "PatchOrchestratorDefaultMaintenanceWindowDuration": 4,
  "PatchOrchestratorDefaultMaintenanceWindowSchedule": "cron(0 18 * * ? *)",
  "PatchOrchestratorDefaultMaintenanceWindowTimeZone": "UTC",
  "PatchOrchestratorDefaultPatchBackupRetentionInDays": 60,
  "PatchOrchestratorNotificationEmails": ["user@test.com"]
}
```

Management Account | Create Custom OUs

Create multiple custom AWS organizational units (OU) under the following paths, "customer-managed", "applications:managed", "applications:tools" and "applications:development".

Full classification: Deployment | Managed landing zone | Management account | Create custom OUs

Change Type Details

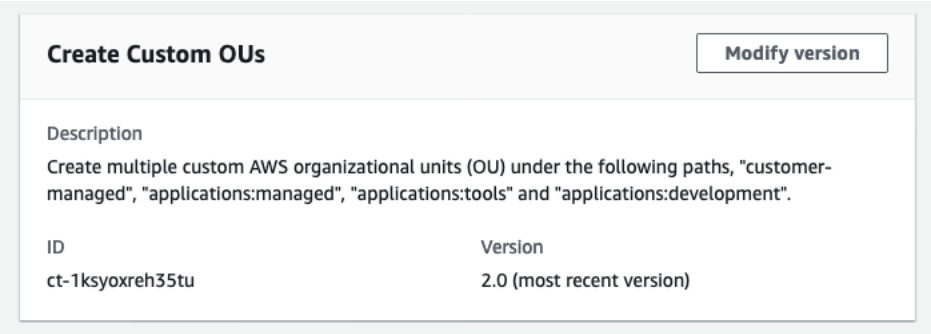
Change type ID	ct-1ksyoxreh35tu
Current version	2.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create a custom OU

Management account: Creating a Management account custom OU with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating a Management account custom OU with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1ksyoxreh35tu" \
--change-type-version "2.0" --title "New OU Creation" \
--execution-parameters "{\"CustomOUPaths\": [ \"applications:managed:OU1:OU2:OU3\",
 \"applications:managed:OU1:OU2:OU3\"]}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it MgmtAcctCreateOuParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1ksyoxreh35tu"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    MgmtAcctCreateOuParams.json
   ```

2. Modify and save the MgmtAcctCreateOuParams file. For example, you can replace the contents with something like this:

   ```
   {
     "CustomOUPaths": ["applications:managed:healthcare", "customer-managed:CustomOU",
    "applications:tools:automation", "applications:development:healthcare"]
   }
   ```

3. Output the RFC template JSON file to a file; this example names it MgmtAcctCreateOuRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > MgmtAcctCreateOuRfc.json
   ```

4. Modify and save the MgmtAcctCreateOuRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-1ksyoxreh35tu",
"Title":                "Management-Acct-Create-OU-RFC"
}
```

5. Create the RFC, specifying the MgmtAcctCreateOu Rfc file and the MgmtAcctCreateOuParams file:

```
aws amscm create-rfc --cli-input-json file://MgmtAcctCreateOuRfc.json  --execution-
parameters file://MgmtAcctCreateOuParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 2.0. A new parameter, **CustomOUPath** replaces the previous **CustomOUName** parameter, and the change type is now automated and not manually executed.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

For information on creating OUs, see Managing organizational units (OUs).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1ksyoxreh35tu.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Management Account | Create Custom SCP (Review Required)

Create a custom service control policy (SCP) to manage permissions across AWS organization.

Full classification: Deployment | Managed landing zone | Management account | Create custom SCP (review required)

Change Type Details

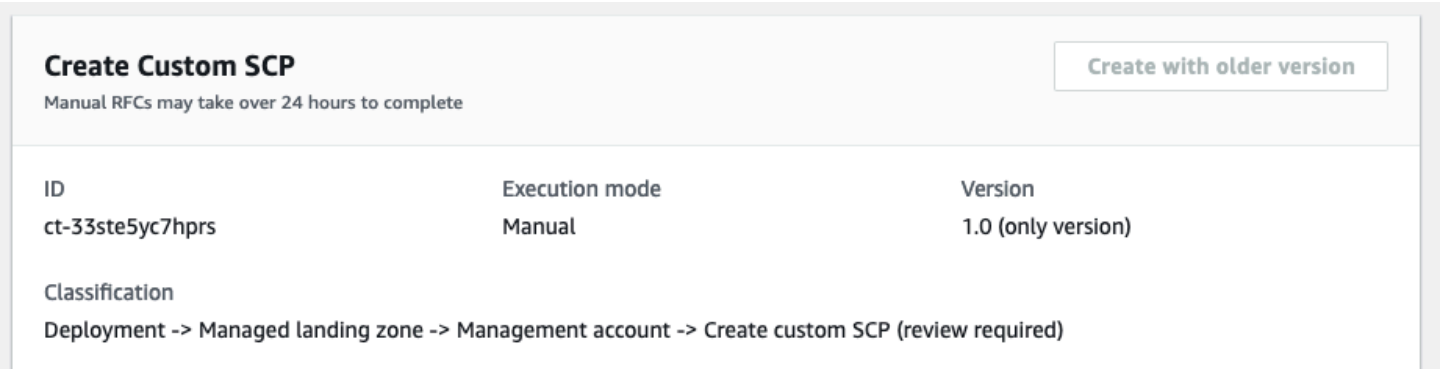
Change type ID	ct-33ste5yc7hprs
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create a service control policy (SCP) (review required)

Management account: Creating a Management account custom SCP with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating a Management account custom SCP with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-33ste5yc7hprs" \
--change-type-version "1.0" --title "New SCP Creation" \
--execution-parameters "{\"TargetId\":\"ou-hlzm-8ievlm9x\",
 \"CustomServiceControlPolicy\":\"Test\", \"SCPDescription\":\"Test SCP\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateMasterAcctScpParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-33ste5yc7hprs"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateMasterAcctScpParams.json
```

2. Modify and save the CreateMasterAcctScpParams file. For example, you can replace the contents with something like this:

```
{
"TargetId":"ou-hlzm-8ievlm9x",
"CustomServiceControlPolicy":"MySCP",
"SCPDescription":"Test SCP"
}
```

3. Output the RFC template JSON file to a file; this example names it CreateMasterAcctScpRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateMasterAcctScpRfc.json
```

4. Modify and save the CreateMasterAcctScpRfc file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-33ste5yc7hprs",
  "ChangeTypeVersion": "1.0",
  "Title": "New SCP Creation"
}
```

5. Create the RFC, specifying the CreateMasterAcctCreateScp Rfc file and the CreateMasterAcctScpParams.json file:

```
aws amscm create-rfc --cli-input-json file://CreateMasterAcctScpRfc.json  --
execution-parameters file://CreateMasterAcctScpParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

> **Note**
>
> Make sure that you refer to and use the curated Service Control Policies (SCPs) library that fits your business requirements. Provide the unique ID from the library in the form of `SCP-AMS-XXX` in the RFC title.
> For more information, see Curated SCPs and Config Rules.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-33ste5yc7hprs.

Example: Required Parameters

```
{
  "TargetId": "ou-96dv-e18n036l",
  "CustomServiceControlPolicy": ""
}
```

Example: All Parameters

```
{
  "TargetId": "ou-96dv-e18n036l",
  "CustomServiceControlPolicy": "",
  "SCPDescription": "Description of the custom Service Control Policy (SCP) that needs
 to be attached to the provided target.",
  "Priority": "Medium"
```

```
}
```

Management Account | Create Customer-Managed Application Account

Create a customer-managed application account in a multi-account AWS landing zone. Customer-managed accounts give you full control to operate the infrastructure within the centralized architecture managed by AMS. Multi-account AWS landing zone core accounts must already be onboarded to AWS Managed Services (AMS).

Full classification: Deployment | Managed landing zone | Management account | Create customer-managed application account

Change Type Details

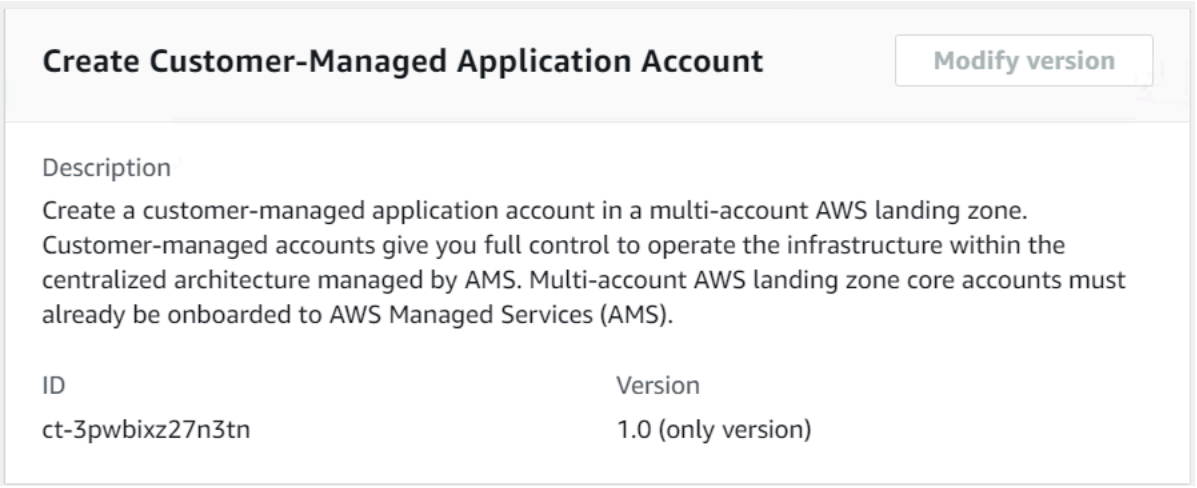
Change type ID	ct-3pwbixz27n3tn
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create a Customer Managed application account

Management account: Creating a Management account customer-managed application account with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating a Management account customer-managed application account with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-3pwbixz27n3tn" \
--change-type-version "1.0" --title "New customer-managed account creation" \
--execution-parameters "{\"AccountName\":\"test\", \"AccountEmail\":\"test@test.com\",
 \"CustomerManagedOUName\":\"customer-managed\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it NewCustomerManagedAccountParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1zdasmc2ewzrs"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    NewCustomerManagedAccountParams.json
   ```

2. Modify and save the NewCustomerManagedAccountParams file. For example, you can replace the contents with something like this:

   ```
   {
   "AccountName":"test",
   "AccountEmail":"test@test.com",
   "CustomerManagedOUName":"customer-managed"
   }
   ```

3. Output the RFC template JSON file to a file; this example names it NewCustomerManagedAccountRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > NewCustomerManagedAccountRfc.json
   ```

4. Modify and save the NewCustomerManagedAccountRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeId": "ct-3pwbixz27n3tn",
     "ChangeTypeVersion": "1.0",
     "Title": "New customer-managed account creation"
   ```

```
}
```

5. Create the RFC, specifying the NewCustomerManagedAccount Rfc file and the NewCustomerManagedAccountParams file:

```
aws amscm create-rfc --cli-input-json file://NewCustomerManagedAccountRfc.json  --
execution-parameters file://NewCustomerManagedAccountParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3pwbixz27n3tn.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Management Account | Create Developer Mode Account (With VPC)

Create a managed AWS landing zone developer mode account and a VPC with up to 10 private subnets and up to 5 optional public subnets per availability zone (AZ) for two or three AZ's. Optionally, also create an AWS Backup plan with up to four different rules. Managed AWS landing zone core accounts must already be onboarded to AWS Managed Services (AMS).

Full classification: Deployment | Managed landing zone | Management account | Create developer mode account (with VPC)

Change Type Details

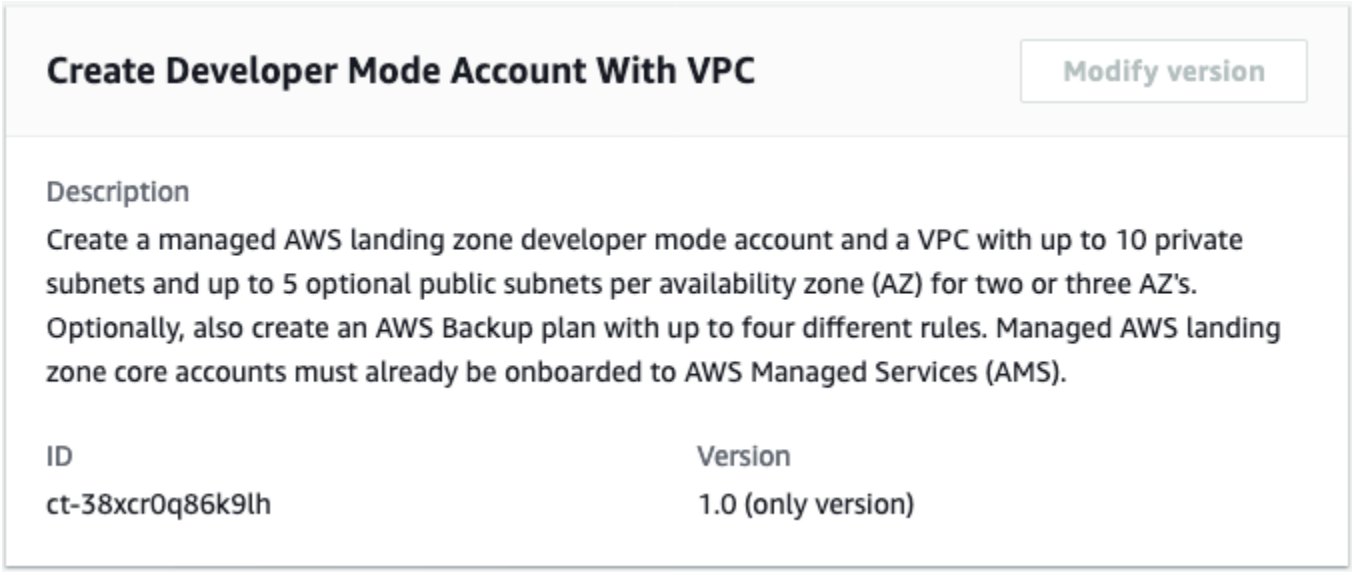
Change type ID	ct-38xcr0q86k9lh
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create Developer mode account with VPC

Management account: Creating a developer mode account with VPC with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating a developer mode account with VPC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-38xcr0q86k9lh" --change-type-version "1.0" --
title "Dev Mode account onboarding" --execution-parameters "{\"AccountName
    \": \"ACCOUNT_NAME\",\"AccountEmail\": \"/\",\"DeveloperModeOUName\":
    \"Development_OU_NAME:CHILD_OU_NAME\",\"SupportLevel\": \"LEVEL\",\"VpcName\":
    \"VPC_NAME\",\"NumberOfAZs\": \"INTEGER\",\"VpcCIDR\":
    \"X.X.X.X/X\", \"PrivateSubnet1AZ1CIDR\": \"X.X.X.X/X\",\"PrivateSubnet1AZ2CIDR\":
    \"X.X.X.X/X\",\"PrivateSubnet1AZ3CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ1CIDR\":
    \"X.X.X.X/X\",\"PublicSubnetAZ2CIDR\": \"X.X.X.X/X\",\"PublicSubnetAZ3CIDR\":
    \"X.X.X.X/X\", \"RouteType\": \"ROUTE_TYPE\",
 \"TransitGatewayApplicationRouteTableName\":
    \"TABLE_NAME\", \"BackupPlanName\":\"PLAN_NAME\", \"ResourceTagKey\":
    \"TAG_KEY\", \"ResourceTagValue\":\"TAG_VALUE\", "\BackupRule1ScheduleExpression\":
    \"cron(0 2 ? * * *)\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateDevModeAcctWithVpcParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-38xcr0q86k9lh"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateDevModeAcctWithVpcParams.json
```

2. Modify and save the CreateDevModeAcctWithVpcParams file. For example, you can replace the contents with something like this:

```
{
      "AccountName": "ACCOUNT_NAME",
      "AccountEmail": "ACCOUNT_EMAIL",
      "DeveloperModeOUName": "DEVELOPER_MODE_OU_NAME:CHILD_OU_NAME",
      "SupportLevel": "PLUS_or_PREMIUM",
      "VpcName": "VPC_NAME",
      "NumberOfAZs": "TWO_or_THREE",
      "VpcCIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ1CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ2CIDR": "x.x.x.x/x",
      "PrivateSubnet1AZ3CIDR": "x.x.x.x/x",
      "PublicSubnetAZ1CIDR": "x.x.x.x/x",
      "PublicSubnetAZ2CIDR": "x.x.x.x/x",
      "PublicSubnetAZ3CIDR": "x.x.x.x/x",
      "RouteType": "ROUTABLE_or_ISOLATED",
      "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME"
}
```

3. Output the RFC template JSON file to a file; this example names it CreateDevModeAcctWithVpcRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateDevModeAcctWithVpcRfc.json
```

4. Modify and save the CreateDevModeAcctWithVpcRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-38xcr0q86k9lh",
  "ChangeTypeVersion": "1.0",
```

```
  "Title": "Create developer mode account with VPC"
}
```

5. Create the RFC, specifying the CreateDevModeAcctWithVpcRfc file and the CreateDevModeAcctWithVpcParams file:

```
aws amscm create-rfc --cli-input-json file://CreateDevModeAcctWithVpcRfc.json  --
execution-parameters file://CreateDevModeAcctWithVpcParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about developer mode, see Developer mode.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-38xcr0q86k9lh.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Management Account | Create StackSets Stack (Review Required)

Create AWS CloudFormation (CFN) StackSets stacks and deploy the stack instances. Use the CloudFormation StackSets feature to create stacks across multiple accounts.

Full classification: Deployment | Managed landing zone | Management account | Create StackSets stack (review required)

Change Type Details

Change type ID	ct-16pknsfa8lul7

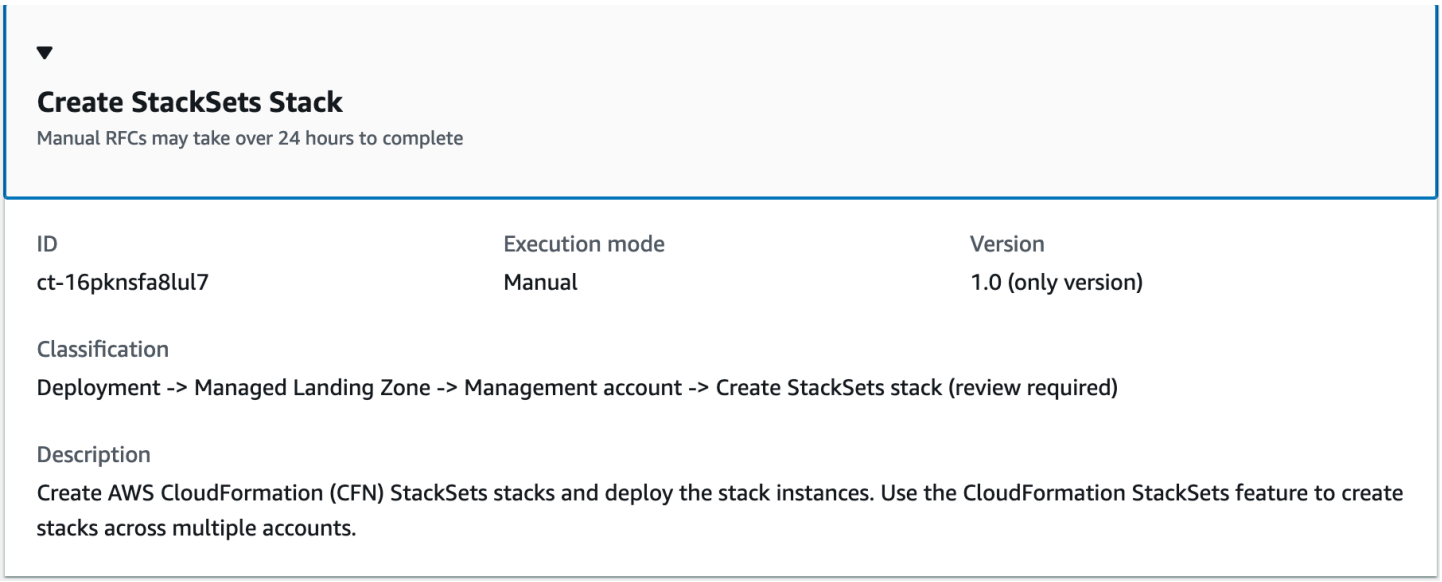
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create a Stacksets stack

Creating a Stacksets stack with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Stacksets stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-16pknsfa8lul7" --change-type-version "1.0"
 --title "Create StackSets Stack" --execution-parameters "{\"Name\": \"Stackset name\",
 \"Region\": \"us-east-1\", \"OuId\": \"ou-cccc-00000000\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateStacksetsStackParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1v9g9n30woc8h"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateStacksetsStackParams.json
   ```

2. Modify and save the UpdateStacksetsStackParams file. For example, you can replace the contents with something like this:

```
{
   "CloudFormationTemplate": "template",
   "CloudFormationTemplateS3Endpoint": "S3 link of the template",
   "Description": "Create Stackset",
   "Name": "test-stackset",
   "OuId": ["ou-cccc-00000000"],
   "Region": "us-east-1",
   "Parameters": [
     { "Name": "test-value",
       "Value": "test-value" }
   ],
   "Tags": [
     {
       "Key": "key1",
       "Value": "value1"
     },
     {
       "Key": "key2",
       "Value": "value2"
     }
   ],
   "Priority": "High"
 }
```

3. Output the RFC template JSON file to a file; this example names it UpdateStacksetsStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateStacksetsStackRfc.json
```

4. Modify and save the UpdateStacksetsStackRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-16pknsfa8lul7",
  "Title": "Create StackSets Stack "
}
```

5. Create the RFC, specifying the UpdateStacksetsStack Rfc file and the UpdateStacksetsStackParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateStacksetsStackRfc.json  --
execution-parameters file://UpdateStacksetsStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For AWS CloudFormation details, see Create a stack set
- For general AWS CloudFormation information on stack sets, see StackSets concepts
- To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-16pknsfa8lul7.

Example: Required Parameters

```
{
  "Description": "AMSTestCT - Create a test stackset",
  "Name": "test-stackset",
  "OuId": ["ou-cccc-00000000"],
  "Region": "us-east-1"
}
```

Example: All Parameters

```
{
  "CloudFormationTemplate": "template",
  "CloudFormationTemplateS3Endpoint": "https://s3.amazonaws.com/cf-
templates-33kj7hiuwdk9-us-east-1/2017261mYA-stm-dynamic-sqs-no-params-
sept-2017.template",
  "Description": "AMSTestCT - Create a test stackset",
  "Name": "test-stackset",
  "OuId": ["ou-cccc-00000000"],
  "Region": "us-east-1",
```

```
  "Parameters": [
    { "Name": "test-value",
      "Value": "test-value" }
  ],
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "Priority": "High"
}
```

Management Account | Create Tools Account (With VPC)

Create a managed AWS landing zone tools account and a VPC with a private subnet, an isolated private subnet, and a public subnet. Optionally, also create an AWS Backup plan with up to four different rules. Managed AWS landing zone core accounts must already be onboarded to AWS Managed Services (AMS).

Full classification: Deployment | Managed landing zone | Management account | Create tools account (with VPC)

Change Type Details

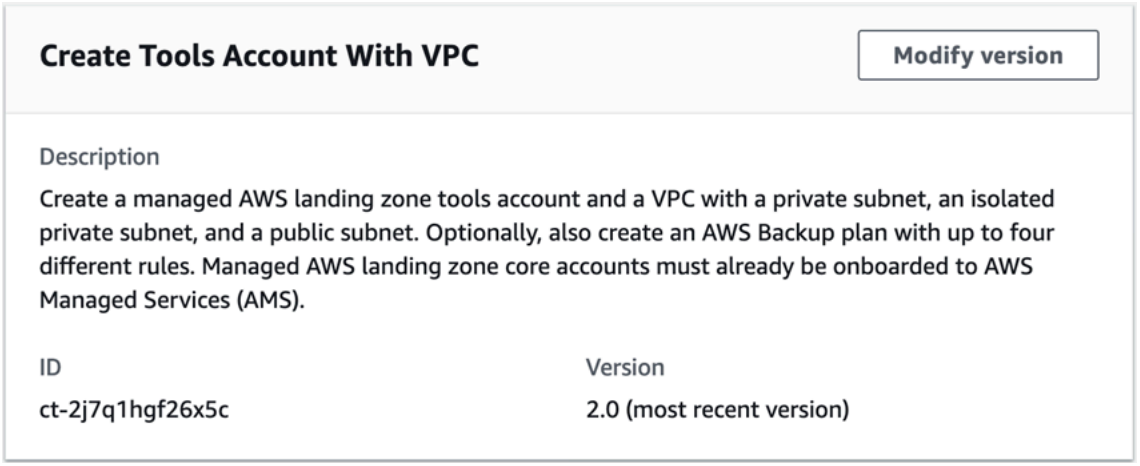
Change type ID	ct-2j7q1hgf26x5c
Current version	2.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create Tools account with VPC

Management account: Creating a Management account Tools account with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Creating a Management account Tools account with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-2j7q1hgf26x5c" \
--change-type-version "1.0" --title "New tools account creation" \
--execution-parameters "{\"AccountName\": \"tools\",\"AccountEmail
\": \"test@test.com\",\"ApplicationOUName\": \"applications:tools\",
\"TransitGatewayApplicationRouteTableName\": \"defaultAppRouteDomain\",
\"SupportLevel\": \"plus\",\"VpcName\": \"testvpc4\",\"VpcCIDR\": \"10.106.0.0/24\",
 \"PrivateSubnetIsolatedCIDR\": \"10.106.0.128/26\", \"PrivateSubnetCIDR\":
\"10.106.0.192/26\",\"PublicSubnetCIDR\":\"10.106.0.192/26\",\"DirectAlertsEmail
\": \"test@test.com\",\"BackupRule1ScheduleExpression\": \"cron(0 2 ? * * )\",
\"BackupPlanName\": \"test\",\"ResourceTagKey\": \"backup\",\"ResourceTagValue\":
 \"true\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it NewToolsAccountParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2j7q1hgf26x5c"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    NewToolsAccountParams.json
   ```

2. Modify and save the NewToolsAccountParams file. For example, you can replace the contents with something like this:

   ```
   {
   "AccountName":"tools",
   "AccountEmail":"test@test.com",
   "ApplicationOUName":"applications:tools",
   "TransitGatewayApplicationRouteTableName": "defaultAppRouteDomain",
   ```

```
"SupportLevel": "plus",
"VpcName": "testvpc4",
"VpcCIDR": "10.106.0.0/24",
"PrivateSubnetIsolatedCIDR": "10.106.0.128/26",
"PrivateSubnetCIDR":"10.106.0.192/26",
"PublicSubnetCIDR":"10.106.0.192/26",
"DirectAlertsEmail": "test@test.com",
"BackupRule1ScheduleExpression": "cron(0 2 ? * * )",
"BackupPlanName": "test",
"ResourceTagKey": "backup",
"ResourceTagValue": "true"
}
```

3. Output the RFC template JSON file to a file; this example names it NewToolsAccountRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > NewToolsAccountRfc.json
```

4. Modify and save the NewToolsAccountRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-2j7q1hgf26x5c",
  "ChangeTypeVersion": "2.0",
  "Title": "New tools account with VPC creation"
}
```

5. Create the RFC, specifying the NewToolsAccount Rfc file and the NewToolsAccountParams file:

```
aws amscm create-rfc --cli-input-json file://NewToolsAccountRfc.json  --execution-
parameters file://NewToolsAccountParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is updated to version 2.0 with changes to input parameters.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2j7q1hgf26x5c.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Networking Account | Add Static Route

Create a static route on transit gateway (TGW) route table. Use this change type for multi-account landing zone (MALZ) Networking accounts only.

Full classification: Deployment | Managed landing zone | Networking account | Add static route

Change Type Details

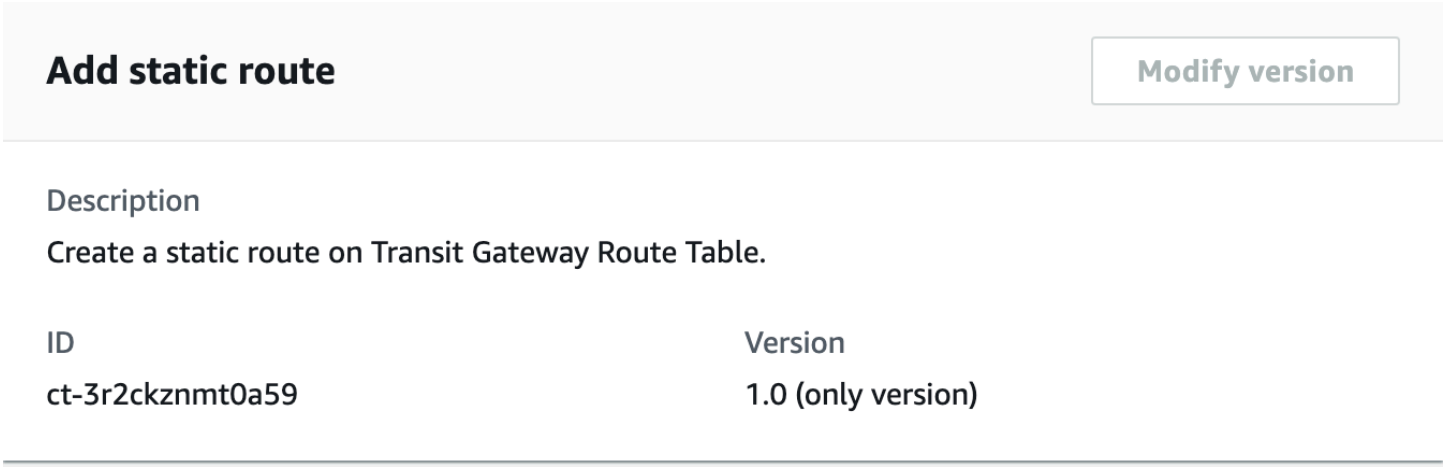
Change type ID	ct-3r2ckznmt0a59
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add a static route

Networking account: Adding a static route with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Adding a static route with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3r2ckznmt0a59" --change-type-version
 "1.0" --title "Create a static route on Transit Gateway Route Table" --execution-
parameters "{\"DocumentName\": \"AWSManagedServices-CreateRouteInTGWRouteTable
\",\"Region\": \"us-east-1\",\"Parameters\": {\"TransitGatewayAttachmentId\":
 [\"tgw-attach-0878cf82a40721d19\"],\"TransitGatewayRouteTableId\": [\"tgw-
rtb-06ddc751c0c0c881c\"], \"Blackhole\": [\"false\"], \"DestinationCidrBlock\":
 [\"10.0.0.0/24\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AddStaticRouteParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3r2ckznmt0a59" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddStaticRouteParams.json
   ```

2. Modify and save the AddStaticRouteParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateRouteInTGWRouteTable",
     "Region": "us-east-1",
     "Parameters": {
       "DestinationCidrBlock" : [ "10.0.0.0/24" ],
       "Blackhole" : [ "false" ],
       "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
       "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it AddStaticRouteRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AddStaticRouteRfc.json
   ```

4. Modify and save the AddStaticRouteRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3r2ckznmt0a59",
  "Title": "Create a static route on Transit Gateway Route Table"
}
```

5. Create the RFC, specifying the AddStaticRouteRfc file and the AddStaticRouteParams file:

```
aws amscm create-rfc --cli-input-json file://AddStaticRouteRfc.json  --execution-
parameters file://AddStaticRouteParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Before you run this Change Type, confirm the following points:

- The TGW route table exists and is available.
- The TGW route table is not `DMZBastionsRouteDomain` or `EgressRouteDomain`.
- The TGW attachment exists.
- The CIDR is not default (0.0.0.0/0), invalid, or that the route already exists.

Note

If you want to add a route in the `DMZBastionsRouteDomain` or `EgressRouteDomain` route table, then use the ct-0xdawir96cy7k to open a MOO RFC.

Note

This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3r2ckznmt0a59.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateRouteInTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "DestinationCidrBlock": ["10.0.2.0/24"],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateRouteInTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "Blackhole": [false],
    "DestinationCidrBlock": ["10.0.2.0/24"],
    "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Networking Account | Create Application Route Table (Review Required)

Create a custom AWS Transit Gateway (TGW) route table for the application accounts in the networking account. By default, the route table does not connect to the on-premise network, but contains preset routes. To request connections to the on-premise network, submit a Management|Other|Other|Update change type.

Full classification: Deployment | Managed landing zone | Networking account | Create application route table (review required)

Change Type Details

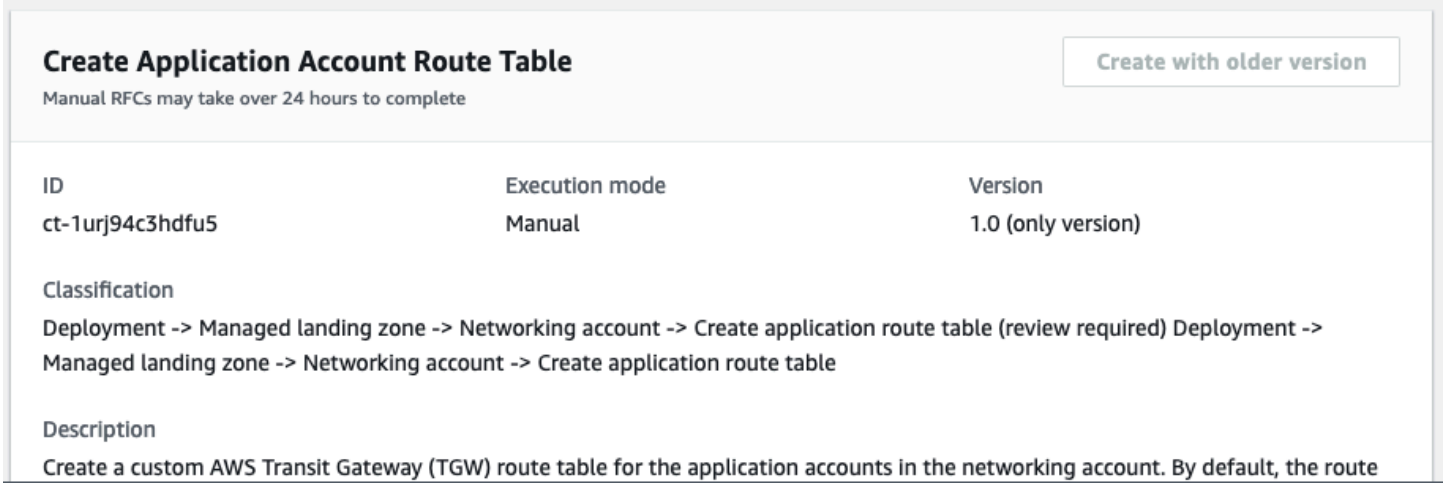
Change type ID	ct-1urj94c3hdfu5
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create application route table (review required)

Networking account: creating an application route table with the Console (review required)

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: creating an application route table with the CLI (review required)

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1urj94c3hdfu5" --change-type-version
 "1.0" --title "Create Application TGW route table" --execution-parameters
 "{\"TransitGatewayApplicationRouteTableName\":\"TABLE_NAME\", \"AddPresetStaticRoutes
\": true}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateRouteTableParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1urj94c3hdfu5"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRouteTableParams.json
   ```

2. Modify and save the CreateRouteTableParams file. For example, you can replace the contents with something like this:

   ```
   {
       "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME",
           "AddPresetStaticRoutes": true
   ```

```
}
```

3. Output the RFC template JSON file to a file; this example names it CreateRouteTableRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRouteTableRfc.json
```

4. Modify and save the CreateRouteTableRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1urj94c3hdfu5",
"Title":                "Create-TG-Route-Table-RFC"
}
```

5. Create the RFC, specifying the CreateRouteTableRfc file and the CreateRouteTableParams file:

```
aws amscm create-rfc --cli-input-json file://CreateRouteTableRfc.json  --execution-
parameters file://CreateRouteTableParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

- This change type is manual. To use the automated version of this change type, see Networking Account | Create Application Route Table.
- By default, the route table does not connect to on-premise network, but contains preset routes. To request connections to the on-premise network, submit a Deployment | Managed landing zone | Networking account | Add static route change type, with the route table ID, to add routes to it.

If you set the **AddPresetStaticRoutes** parameter to False, the route table that created is empty and you must file a Deployment | Managed landing zone | Networking account | Add static route change type, with the route table ID, to add routes to it.

- To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1urj94c3hdfu5.

Example: Required Parameters

```
{
  "TransitGatewayApplicationRouteTableName": "routeTableName"
}
```

Example: All Parameters

```
{
  "TransitGatewayApplicationRouteTableName": "routeTableName",
  "AddPresetStaticRoutes": true,
  "Priority": "Medium"
}
```

Networking Account | Create Transit Gateway Route Table

Create a transit gateway (TGW) route table. Use this change type for multi-account landing zone (MALZ) Networking accounts only.

Full classification: Deployment | Managed landing zone | Networking account | Create transit gateway route table

Change Type Details

Change type ID	ct-3dscwaeyi6cup
Current version	1.0

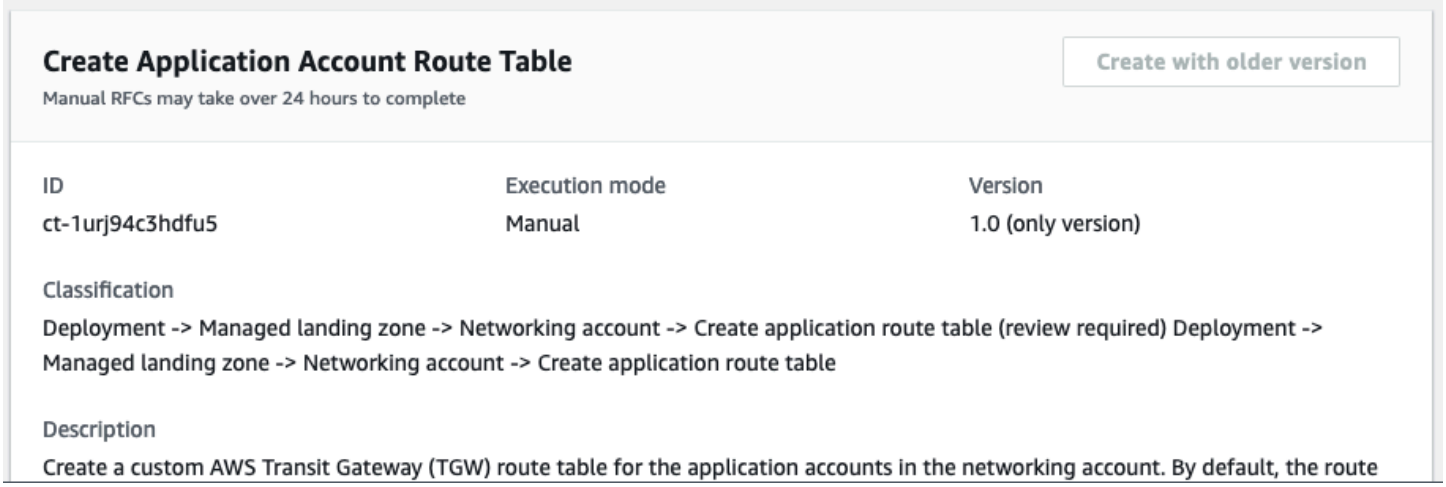
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create application route table (review required)

Networking account: creating an application route table with the Console (review required)

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: creating an application route table with the CLI (review required)

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1urj94c3hdfu5" --change-type-version
 "1.0" --title "Create Application TGW route table" --execution-parameters
 "{\"TransitGatewayApplicationRouteTableName\":\"TABLE_NAME\", \"AddPresetStaticRoutes
\": true}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateRouteTableParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1urj94c3hdfu5"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRouteTableParams.json
   ```

2. Modify and save the CreateRouteTableParams file. For example, you can replace the contents with something like this:

   ```
   {
       "TransitGatewayApplicationRouteTableName": "ROUTE_TABLE_NAME",
           "AddPresetStaticRoutes": true
   }
   ```

3. Output the RFC template JSON file to a file; this example names it CreateRouteTableRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRouteTableRfc.json
```

4. Modify and save the CreateRouteTableRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1urj94c3hdfu5",
"Title":                "Create-TG-Route-Table-RFC"
}
```

5. Create the RFC, specifying the CreateRouteTableRfc file and the CreateRouteTableParams file:

```
aws amscm create-rfc --cli-input-json file://CreateRouteTableRfc.json  --execution-
parameters file://CreateRouteTableParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

- This change type is manual. To use the automated version of this change type, see Networking Account | Create Application Route Table.
- By default, the route table does not connect to on-premise network, but contains preset routes. To request connections to the on-premise network, submit a Deployment | Managed landing zone | Networking account | Add static route change type, with the route table ID, to add routes to it.

  If you set the **AddPresetStaticRoutes** parameter to False, the route table that created is empty and you must file a Deployment | Managed landing zone | Networking account | Add static route change type, with the route table ID, to add routes to it.

- To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3dscwaeyi6cup.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayRouteTableName": "NewApplicationRouteTable1",
    "TransitGatewayId": "tgw-0123456789abcdefg",
    "TGWRouteTableType": "createApplicationRouteDomain"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayRouteTableName": "NewApplicationRouteTable",
    "TransitGatewayId": "tgw-0123456789abcdefg",
    "TGWRouteTableType": "createApplicationRouteDomain"
  }
}
```

Monitoring and Notification Subcategory

Change Type Items and Operations in the Monitoring and Notification Subcategory

- CloudWatch | Create Alarms
- CloudWatch | Create LogGroup
- GuardDuty IP Set | Create (Review Required)
- GuardDuty Threat Intel Set | Create (Review Required)

- SNS | Create (Topic and Subscription)
- SQS | Create

CloudWatch | Create Alarms

Create one or more CloudWatch alarms. For detailed information on CloudWatch alarm properties, see AWS documentation "Creating CloudWatch Alarms".

Full classification: Deployment | Monitoring and notification | CloudWatch | Create alarms

Change Type Details

Change type ID	ct-361vpyun9a9dd
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CloudWatch alarm

Creating a CloudWatch alarm with the console

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CloudWatch alarm with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-361vpyun9a9dd" --change-type-version "1.0" --title "Test Create CloudWatch Alarms"
 --execution-parameters "{\"Alarms\": [{\"ActionsEnabled\": true,\"AlarmActions\":
 [\"arn:aws:sns:us-east-1:000000000000:SNS-Topic\"],\"AlarmDescription\": \"Test
 alarm description.\",\"AlarmName\": \"Test alarm name\",\"ComparisonOperator\":
 \"GreaterThanThreshold\",\"DatapointsToAlarm\": 1,\"Dimensions\": [{\"Name\":
 \"InstanceId\",\"Value\": \"i-12345678901234567\"}],\"EvaluateLowSampleCountPercentile
\": \"ignore\",\"EvaluationPeriods\": 2,\"InsufficientDataActions\": [\"arn:aws:sns:us-
east-1:000000000000:SNS-Topic\"],\"MetricName\": \"TestMetric\",\"Namespace\":
 \"AWS/Test\",\"OkActions\": [\"arn:aws:sns:us-east-1:000000000000:SNS-Topic\"],
\"Period\": 300,\"Statistic\": \"Average\",\"Threshold\": 85,\"TreatMissingData\":
 \"breaching\",\"Unit\": \"Percent\"}],\"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CwAlarmsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-361vpyun9a9dd" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CwAlarmsParams.json
   ```

2. Modify and save the CwAlarmsParams.json file. For example, you can replace the contents with something like this:

```
{
  "Alarms": [
    {
      "ActionsEnabled": true,
      "AlarmActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "AlarmDescription": "Test alarm description.",
      "AlarmName": "Test alarm name",
      "ComparisonOperator": "GreaterThanThreshold",
      "DatapointsToAlarm": 1,
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-12345678901234567"
        }
      ],
      "EvaluateLowSampleCountPercentile": "ignore",
      "EvaluationPeriods": 2,
      "InsufficientDataActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "MetricName": "TestMetric",
      "Namespace": "AWS/Test",
      "OkActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "Period": 300,
      "Statistic": "Average",
      "Threshold": 85,
      "TreatMissingData": "breaching",
      "Unit": "Percent"
    }
  ],
  "Region": "us-east-1"
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it CwAlarmsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CwAlarmsRfc.json
```

4. Modify and save the CwAlarmsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-361vpyun9a9dd",
```

```
"Title":                "CW-ALARMS-RFC"
}
```

5. Create the RFC, specifying the CwAlarmsRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CwAlarmsRfc.json --execution-
parameters file://CwAlarmsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about CloudWatch, see Creating Amazon CloudWatch Alarms.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-361vpyun9a9dd.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Alarms": [
    {
      "ActionsEnabled": true,
      "AlarmActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "AlarmDescription": "Test alarm description.",
      "AlarmName": "Test alarm name",
      "ComparisonOperator": "GreaterThanThreshold",
      "DatapointsToAlarm": 1,
      "Dimensions": [
        {
          "Name": "InstanceId",
          "Value": "i-12345678901234567"
        }
      ],
```

```
      "EvaluateLowSampleCountPercentile": "ignore",
      "EvaluationPeriods": 2,
      "InsufficientDataActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "MetricName": "TestMetric",
      "Namespace": "AWS/Test",
      "OkActions": ["arn:aws:sns:us-east-1:000000000000:SNS-Topic"],
      "Period": 300,
      "Statistic": "Average",
      "Threshold": 85,
      "TreatMissingData": "breaching",
      "Unit": "Percent"
    }
  ],
  "Region": "us-east-1"
}
```

CloudWatch | Create LogGroup

Creates a CloudWatch LogGroup with optional subscription filter, up to 5 log streams and up to 5 metric filters.

Full classification: Deployment | Monitoring and notification | CloudWatch | Create LogGroup

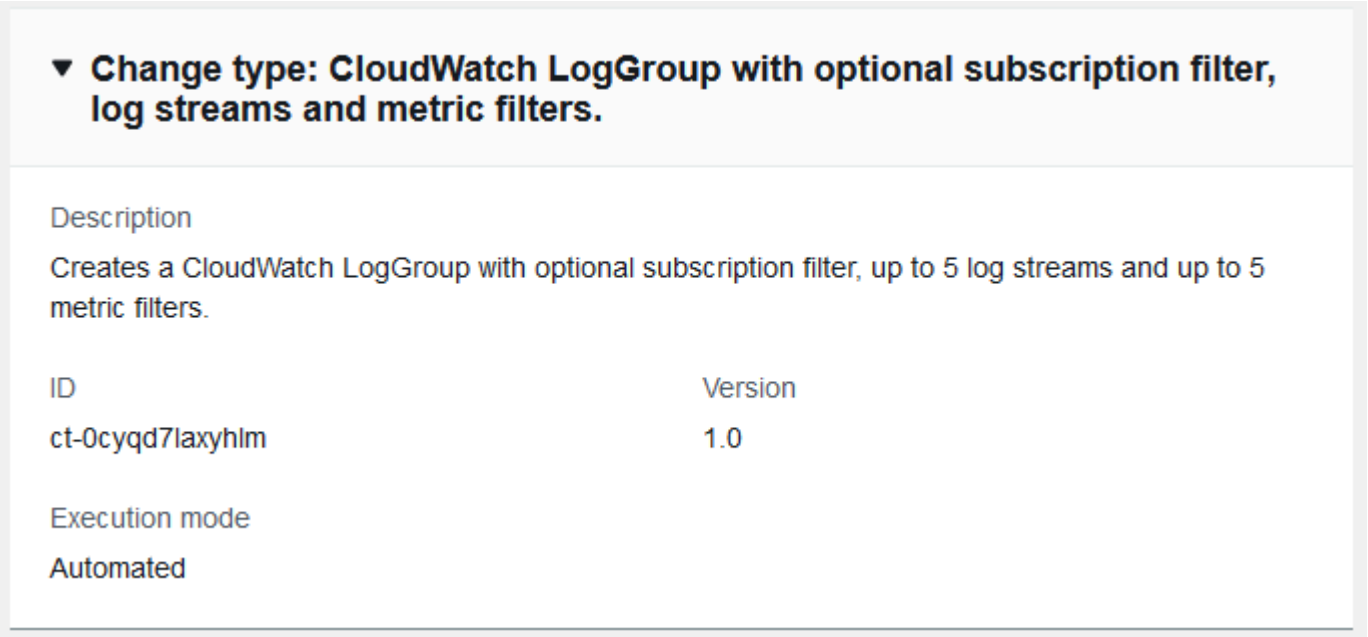
Change Type Details

Change type ID	ct-0cyqd7laxyhlm
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create CloudWatch LogGroup

Creating a CloudWatch LogGroup with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a CloudWatch LogGroup with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-0cyqd7laxyhlm" --change-type-version "1.0" --title 'CloudWatch LogGroup'
 --description "CloudWatch LogGroup"  --execution-parameters "{\"Description
\":\"My Test LogGroup\",\"VpcId\":\"VPC_ID\",\"Name\":\"Test LogGroup\",
\"StackTemplateId\":\"stm-8ian3plt5a6jbv7jt\",\"TimeoutInMinutes\":60,\"Parameters
\": {\"LogGroupName\":\"customer-testloggroup\",\"LogStream1Name\":\"LogStream1\",
\"SubscriptionFilterPattern\":\"test\",\"SubscriptionDestinationARN\":
\"arn:aws:lambda:us-east-1:123456789012:function:test_lambda\",\"MetricFilter1Name\":
\"test_metric_filter1\",\"MetricFilter1Namespace\":\"test_metric_filter1_namespace\",
\"MetricFilter1Pattern\":\"{$.eventType=\\\"test_event\\\"}\",\"MetricFilter1Value\":
\"10\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CwLGParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-ct-0cyqd7laxyhlm" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CwLGParams.json
   ```

2. Modify and save the CwLGParams.json file. For example, you can replace the contents with something like this:

   ```
   {
       "Description": "Test CloudWatch Description",
       "VpcId": "VPC_ID",
       "StackTemplateId": "stm-8ian3plt5a6jbv7jt",
       "Name": "My_CW_Loggroup",
       "TimeoutInMinutes": 60,
       "Parameters": {
           "LogGroupName": "customer-testloggroup",
   ```

```
        "LogStream1Name": "LogStream1",
        "SubscriptionFilterPattern": "test",
        "SubscriptionDestinationARN": "arn:aws:lambda:us-
east-1:123456789012:function:test_lambda",
        "MetricFilter1Name": "test_metric_filter1",
        "MetricFilter1Namespace": "test_metric_filter1_namespace",
        "MetricFilter1Pattern": "{$.eventType=\"test_event\"}",
        "MetricFilter1Value": "10"
    }
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; example names it CwLGRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CwLGRfc.json
```

4. Modify and save the CwLGRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0cyqd7laxyhlm",
"Title":                "CW-LG-RFC"
}
```

5. Create the RFC, specifying the CwLGRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CwLGRfc.json --execution-parameters
 file://CwLGParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about CloudWatch, see Creating Amazon CloudWatch Alarms.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0cyqd7laxyhlm.

Example: Required Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-8ian3plt5a6jbv7jt",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "LogGroupName": "customer-testloggroup"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-1234567890abcdef0",
  "StackTemplateId": "stm-8ian3plt5a6jbv7jt",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "LogGroupName":"customer-test",
```

```
    "LogGroupRetentionInDays":"7",
    "LogStream1Name":"logstream1",
    "LogStream2Name":"logstream2",
    "LogStream3Name":"logstream3",
    "LogStream4Name":"logstream4",
    "LogStream5Name":"logstream5",
    "SubscriptionFilterIAMroleARN":"arn:aws:iam::123456789012:role/example-role",
    "SubscriptionFilterPattern":"Error",
    "SubscriptionDestinationARN":"arn:aws:kinesis:us-east-1:123456789012:stream/
example-stream-name",
    "MetricFilter1Name":"metricfilter1",
    "MetricFilter1Namespace":"metricfilter1namespace",
    "MetricFilter1Pattern":"Error",
    "MetricFilter1Value":"10",
    "MetricFilter1DefaultValue":"1",
    "MetricFilter2Name":"metricfilter2",
    "MetricFilter2Namespace":"metricfilter2namespace",
    "MetricFilter2Pattern":"Error",
    "MetricFilter2Value":"20",
    "MetricFilter2DefaultValue":"1",
    "MetricFilter3Name":"metricfilter3",
    "MetricFilter3Namespace":"metricfilter3namespace",
    "MetricFilter3Pattern":"Error",
    "MetricFilter3Value":"30",
    "MetricFilter3DefaultValue":"1",
    "MetricFilter4Name":"metricfilter4",
    "MetricFilter4Namespace":"metricfilter4namespace",
    "MetricFilter4Pattern":"40",
    "MetricFilter4Value":"2",
    "MetricFilter4DefaultValue":"1",
    "MetricFilter5Name":"metricfilter5",
    "MetricFilter5Namespace":"metricfilter5namespace",
    "MetricFilter5Pattern":"Error",
    "MetricFilter5Value":"50",
    "MetricFilter5DefaultValue":"1"
  }
}
```

GuardDuty IP Set | Create (Review Required)

Use to create an Amazon GuardDuty IPSet instance which is a list of trusted IP addresses that have been whitelisted for highly secure communication with your AWS environment.

Full classification: Deployment | Monitoring and notification | GuardDuty IP set | Create (review required)

Change Type Details

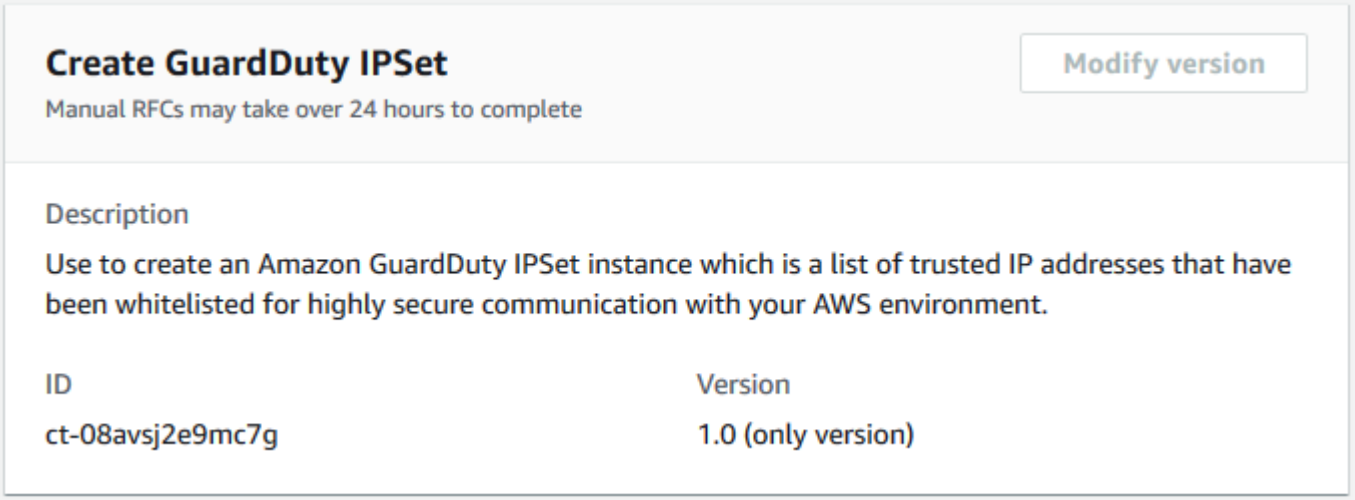
Change type ID	ct-08avsj2e9mc7g
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create GuardDuty IP set (review required)

Creating an IP set for GuardDuty (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an IP set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-08avsj2e9mc7g" --change-type-version "1.0"
 --title "Create Amazon GuardDuty IP Set" --execution-parameters "{\"Activate\": true,
 \"DetectorId\": \"00000000000000000000000000000000\", \"Format\": \"TXT\", \"Name\":
 \"trusted-ips\", \"IpSet\": \"https://s3.us-west-2.amazonaws.com/my-bucket/my-object-
key\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateGdIpSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-08avsj2e9mc7g" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateGdIpSetParams.json
   ```

2. Modify and save the CreateGdIpSetParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "Activate": true,
  "DetectorId": "00000000000000000000000000000000",
  "Format": "TXT",
  "Name": "trusted-ips",
  "IpSet": "https://s3.us-west-2.amazonaws.com/my-bucket/my-object-key",
  "Region": "us-east-1"
}
```

3. Output the RFC template JSON file to a file named CreateGdIpSetRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateGdIpSetRfc.json
```

4. Modify and save the CreateGdIpSetRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-08avsj2e9mc7g",
"Title":                "CREATE_GD_IP_SET"
}
```

5. Create the RFC, specifying the CreateGdIpSet Rfc file and the CreateGdIpSetParams file:

```
aws amscm create-rfc --cli-input-json file://CreateGdIpSetRfc.json  --execution-
parameters file://CreateGdIpSetParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

To learn more about Amazon GuardDuty and creating IP sets, see Amazon GuardDuty and CreateIPSet.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-08avsj2e9mc7g.

Example: Required Parameters

```
{
  "Region": "us-east-1",
  "Name": "Sample IPSet",
  "IpSet": "https://s3.amazonaws.com/guarddutylists/sample.txt"
}
```

Example: All Parameters

```
{
  "Activate": true,
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "Region": "us-east-1",
  "Name": "Sample IPSet",
  "IpSet": "https://s3.amazonaws.com/guarddutylists/sample.txt",
  "Format": "TXT",
  "Priority": "Medium"
}
```

GuardDuty Threat Intel Set | Create (Review Required)

Use to create an Amazon GuardDuty ThreatIntelSet instance, which is a list of known malicious IP addresses that have been blacklisted for communication with your AWS environment.

Full classification: Deployment | Monitoring and notification | GuardDuty threat intel set | Create (review required)

Change Type Details

Change type ID	ct-25v6r7t8gvkq5
Current version	1.0
Expected execution duration	60 minutes

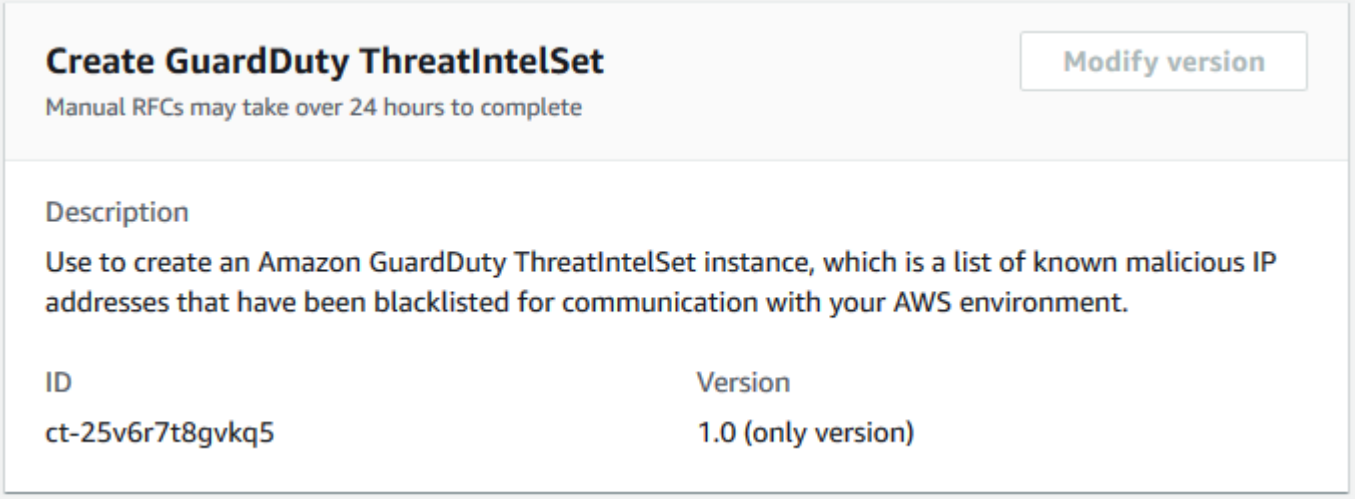
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create GuardDuty Threat intel set (review required)

Creating a Threat intel set for GuardDuty (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a Threat intel set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-25v6r7t8gvkq5" --change-type-version
 "1.0" --title "Create Amazon GuardDuty Threat Intel Set" --execution-parameters
 "{\"Activate\": true, \"DetectorId\": \"00000000000000000000000000000000\", \"Format
\": \"TXT\", \"Name\": \"blacklisted-ips\", \"ThreatIntelSet\": \"https://s3.us-
west-2.amazonaws.com/my-bucket/my-object-key\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateGdThreatIntelSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-25v6r7t8gvkq5"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateGdThreatIntelSetParams.json
   ```

2. Modify and save the CreateGdThreatIntelSetParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Activate": true,
     "DetectorId": "00000000000000000000000000000000",
     "Format": "TXT",
     "Name": "blacklisted-ips",
     "ThreatIntelSet": "https://s3.us-west-2.amazonaws.com/my-bucket/my-object-key",
     "Region": "us-east-1"
   }
   ```

3. Output the RFC template JSON file to a file named CreateGdThreatIntelSetRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateGdThreatIntelSetRfc.json
```

4. Modify and save the CreateGdThreatIntelSetRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-25v6r7t8gvkq5",
"Title":                "CREATE_GD_THREAT_INTEL_SET"
}
```

5. Create the RFC, specifying the CreateGdIpSet Rfc file and the CreateGdThreatIntelSetParams file:

```
aws amscm create-rfc --cli-input-json file://CreateGdThreatIntelSetRfc.json  --
execution-parameters file://CreateGdThreatIntelSetParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

For more information about Amazon GuardDuty and creating Threat Intel sets, see Amazon Guard Duty and CreateThreatIntelSet.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-25v6r7t8gvkq5.

Example: Required Parameters

```
{
```

```
  "Region": "us-east-1",
  "Name": "Sample Threat Intel Set",
  "ThreatIntelSet": "https://s3.amazonaws.com/guarddutylists/sample.txt"
}
```

Example: All Parameters

```
{
  "Activate": true,
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "Region": "us-east-1",
  "Name": "Sample Threat Intel Set",
  "ThreatIntelSet": "https://s3.amazonaws.com/guarddutylists/sample.txt",
  "Format": "TXT",
  "Priority": "Medium"
}
```

SNS | Create (Topic and Subscription)

Create an SNS topic and up to five subscriptions.

Full classification: Deployment | Monitoring and notification | SNS | Create (topic and subscription)

Change Type Details

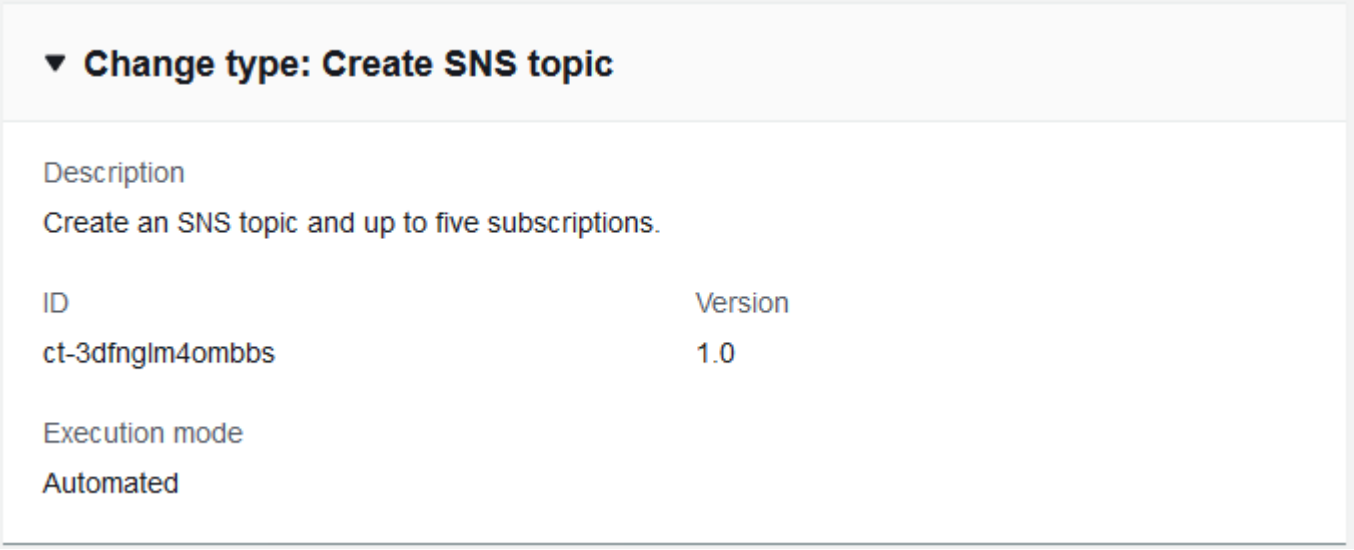
Change type ID	ct-3dfnglm4ombbs
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create SNS topic and subscription

Creating an SNS topic and subscription (up to 5) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an SNS topic and subscription (up to 5) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3dfnglm4ombbs" --change-type-version
 "1.0" --title "CREATE_SNS_TOPIC" --execution-parameters "{\"Description\":
\"SNS_TOPIC_DESCRIPTION\",\"VpcId\":\"VPC_ID\",\"Name\":\"SNS_TOPIC_NAME\",
\"StackTemplateId\":\"stm-eakrsalqo9m62tpun\",\"TimeoutInMinutes\":60,\"Parameters\":
{\"TopicName\":\"mytopic-cli\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateSnsTopicSubParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3dfnglm4ombbs"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateSnsTopicSubParams.json
   ```

2. Modify and save the CreateSnsTopicSubParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "SnsTopicSub-Create",
   "VpcId": "VPC_ID",
   "Name":             "My-SnsTopicSub",
   "Parameters":{
       "TopicName": "mytopic-cli-all-params",
       "DisplayName": "testsns",
       "Subscription1Protocol": "email",
       "Subscription1Endpoint": "abc@xyz.com",
       "Subscription1RawMessageDelivery": "false",
       "Subscription2Protocol": "sms",
       "Subscription2Endpoint": "+61500444777",
       "Subscription2RawMessageDelivery": "false",
   ```

```
    "KmsMasterKeyId": "arn:aws:kms:us-east-1:123456789101:key/
cfe0542d-3be9-4166-9eac-d0cd6af61445"
    }
}
```

3. Output the RFC template JSON file to a file named CreateSnsTopicSubRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateSnsTopicSubRfc.json
```

4. Modify and save the CreateSnsTopicSubRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3dfnglm4ombbs",
"Title":                "SnsTopicSub-Create-RFC"
}
```

5. Create the RFC, specifying the CreateSnsTopicSub Rfc file and the CreateSnsTopicSubParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSnsTopicSubRfc.json  --
execution-parameters file://CreateSnsTopicSubParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Simple Notification Service (SNS) and creating SNS topics and subscriptions, see Amazon Simple Notification Service. Also see Getting Started with Amazon SNS. For pricing information, see Amazon SNS pricing.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3dfnglm4ombbs.

Example: Required Parameters

```
{
```

```
  "Description" : "Creates SNS Topic using random Topic name with no input or parameter
 given.",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-eakrsalqo9m62tpun",
  "TimeoutInMinutes" : 60,
  "Parameters" : { }
}
```

Example: All Parameters

```
{
  "Description" : "Creates just SNS Topic per the given name",
  "VpcId" : "vpc-12345678",
  "Name" : "TestStack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-eakrsalqo9m62tpun",
  "TimeoutInMinutes" : 60,
  "Parameters" : {
    "TopicName" : "MySNSTopic",
    "DisplayName" : "",
    "Subscription1Endpoint" : "",
    "Subscription1RawMessageDelivery" : "false",
    "Subscription2Endpoint" : "",
    "Subscription2RawMessageDelivery" : "false",
    "Subscription3Endpoint" : "",
    "Subscription3RawMessageDelivery" : "false",
    "Subscription4Endpoint" : "",
    "Subscription4RawMessageDelivery" : "false",
    "Subscription5Endpoint" : "",
    "Subscription5RawMessageDelivery" : "false",
    "KmsMasterKeyId" : ""
  }
}
```

SQS | Create

Use to create an Amazon Simple Queue Service instance for messages to be shared by system components.

Full classification: Deployment | Monitoring and notification | SQS | Create

Change Type Details

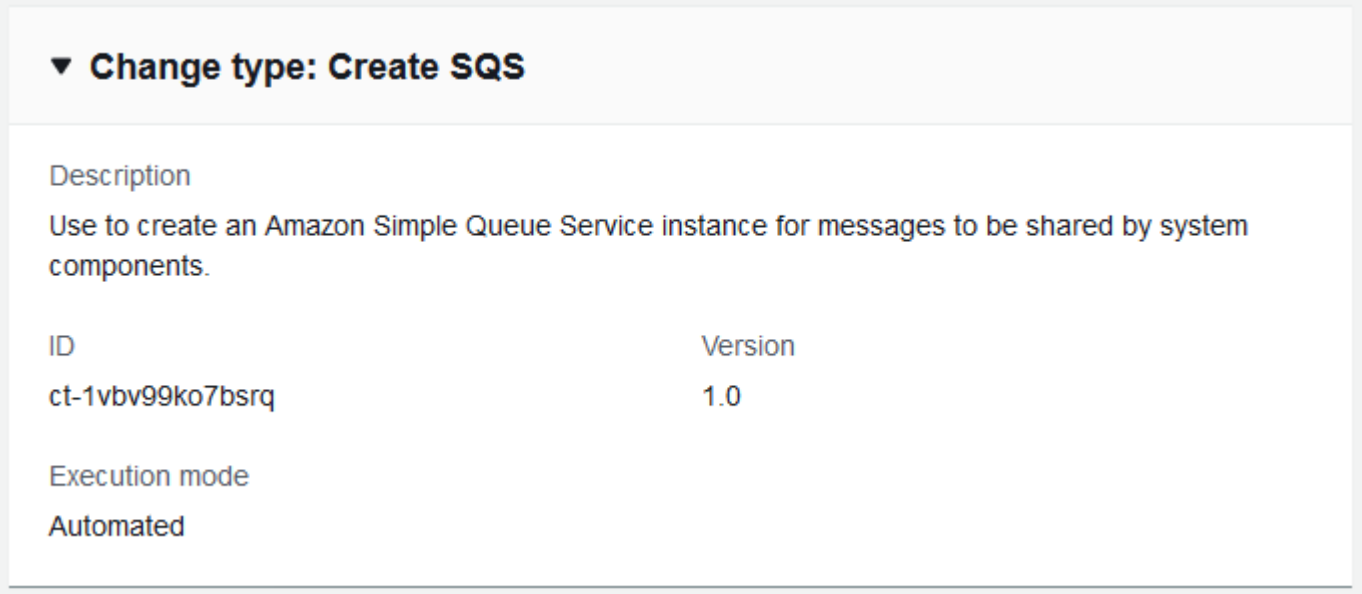
Change type ID	ct-1vbv99ko7bsrq
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create SQS queue

Creating an SQS queue with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an SQS queue with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1vbv99ko7bsrq" --change-type-version "1.0" --
title "Create Amazon SQS Queue" --execution-parameters "{\"Description\": \"SQS-Queue-
Create-RFC\", \"VpcId\": \"VPC_ID\", \"StackTemplateId\": \"stm-s1ejpr80000000000\",
 \"Name\": \"MySqsQueue\", \"Tags\": [{\"Key\": \"my-tag-1\", \"Value\": \"my-tag-
value-1\"}, {\"Key\": \"my-tag-2\", \"Value\": \"my-tag-value-2\"}], \"TimeoutInMinutes
\": 60, \"Parameters\": {\"SQSDelaySeconds\": 0, \"SQSMaximumMessageSize\": 262144,
 \"SQSMessageRetentionPeriod\": 345600, \"SQSQueueName\": \"MyQueueName\",
 \"SQSReceiveMessageWaitTimeSeconds\": 0, \"SQSVisibilityTimeout\": 60}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateSqsInstanceParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vbv99ko7bsrq"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateSqsInstanceParams.json
   ```

2. Modify and save the CreateSqsInstanceParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "SQS-Queue-Create-RFC",
     "VpcId": "VPC_ID",
     "StackTemplateId": "stm-s1ejpr80000000000",
     "Name": "MySqsQueue",
     "Tags": [{
       "Key": "my-tag-1",
       "Value": "my-tag-value-1"
     }, {
       "Key": "my-tag-2",
       "Value": "my-tag-value-2"
     }],
     "TimeoutInMinutes": 60,
     "Parameters": {
       "SQSDelaySeconds": 0,
       "SQSMaximumMessageSize": 262144,
       "SQSMessageRetentionPeriod": 345600,
   ```

```
    "SQSQueueName": "MyQueueName",
    "SQSReceiveMessageWaitTimeSeconds": 0,
    "SQSVisibilityTimeout": 60
  }
}
```

3. Output the RFC template JSON file to a file named CreateSqsInstanceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateSqsInstanceRfc.json
```

4. Modify and save the CreateSqsInstanceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1vbv99ko7bsrq",
"Title":                "Sqs-Instance-Create-RFC"
}
```

5. Create the RFC, specifying the CreateSqsInstance Rfc file and the CreateSqsInstanceParams file:

```
aws amscm create-rfc --cli-input-json file://CreateSqsInstanceRfc.json  --
execution-parameters file://CreateSqsInstanceParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon Simple Queue Service (SQS), see Amazon Simple Queue Service.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1vbv99ko7bsrq.

Example: Required Parameters

```
{
  "Description": "This is a test description",
```

```
  "VpcId": "vpc-01234567890abcdef",
  "StackTemplateId": "stm-s1ejpr80000000000",
  "Name": "Test Stack",
  "TimeoutInMinutes": 60,
  "Parameters": {
    "SQSQueueName": "mytestsqs"
  }
}
```

Example: All Parameters

```
{
  "Description": "This is a test description",
  "VpcId": "vpc-12345678",
  "StackTemplateId": "stm-s1ejpr80000000000",
  "Name": "Test Stack",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "TimeoutInMinutes": 60,
  "Parameters": {
    "SQSDelaySeconds": 0,
    "SQSMaximumMessageSize": 262144,
    "SQSMessageRetentionPeriod": 345600,
    "SQSQueueName": "mytestsqs",
    "SQSReceiveMessageWaitTimeSeconds": 0,
    "SQSVisibilityTimeout": 0
  }
}
```

Patching Subcategory

Change Type Items and Operations in the Patching Subcategory

- SSM Patch Baseline | Create (Amazon Linux 2)
- SSM Patch Baseline | Create (Amazon Linux)

- SSM Patch Baseline | Create (CentOS)
- SSM Patch Baseline | Create (Red Hat)
- SSM Patch Baseline | Create (Windows)
- SSM Patch Window | Create

SSM Patch Baseline | Create (Amazon Linux 2)

Create an AWS Systems Manager (SSM) patch baseline to define which patches are approved for installation on your instances for Amazon Linux 2 OS. Specify existing instance "Patch Group" tag values for the patch baseline. The patch baseline is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch baseline | Create (Amazon Linux 2)

Change Type Details

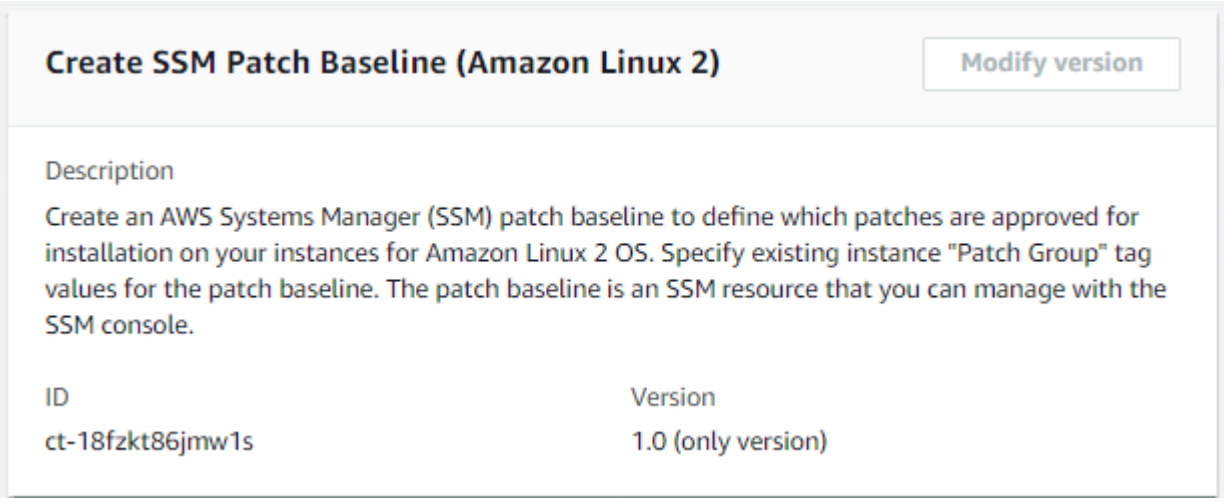
Change type ID	ct-18fzkt86jmw1s
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create for Amazon Linux 2

Creating an SSM Linux2 patch baseline with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

In the AWS Console, you can view the patch baselines you created at Systems Manager --> Patch Manager --> Patch Baselines.

Creating an SSM Linux 2 patch baseline with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title Patch-Baseline-Create-AL2-RFC --change-type-id
 ct-18fzkt86jmw1s --change-type-version 1.0 --execution-parameters ' {"ApprovalRules":
 [{"ApproveAfterDays": 7, "Classification": ["Security"], "Severity": ["All"]}],
 "OperatingSystem": "Amazon Linux 2", "Name": "TestBaselineAmazonLinux2",
 "PatchGroupTagValues": ["MyAmazonLinux2PatchGroup"]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateAL2PatchBaselineParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2taqdgegqthjr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateAL2PatchBaselineParams.json
   ```

2. Modify and save the CreateAL2PatchBaselineParams file, make sure to modify these parameters to meet your specific needs.

   In this example, all critical security updates are approved for installation five days after release. Patches related to vulnerability ID CVE-2017-5754 are approved immediately even if they are not a critical severity. Finally, example-pkg-0.710.10-2.7.abcd.x86_64 will not be installed, even if it matches an approval rule.

   ```
   {
   "ApprovalRules":[{
       "ApproveAfterDays": 5,
       "Severity": [
           "Critical"
           ],
       "Classification": [
           "Security"
           ]
       }],
   "ApprovedPatches":["CVE-2017-5754"],
   "Description": "Patch Baseline",
   "Name": "PatchBaseline-Unit-test",
   "OperatingSystem": "Amazon Linux 2",
   ```

```
"PatchGroupTagValues": [
    "test1"
    ],
"RejectedPatches":["example-pkg-0.710.10-2.7.abcd.x86_64"],
"Tags": [
    {
    "Key":"patchGroupAL2",
    "Value":"test1"
    }
]
```

3. Output the RFC template to a file in your current folder; this example names it CreateAL2PatchBaselineRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAL2PatchBaselineRfc.json
```

4. Modify and save the CreateAL2PatchBaselineRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-18fzkt86jmw1s",
"Title":                "Patch-Baseline-Create-AL2-RFC"
}
```

5. Create the RFC, specifying the CreateAL2PatchBaselineRfc file and the CreateAL2PatchBaselineParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAL2PatchBaselineRfc.json --
execution-parameters file://CreateAL2PatchBaselineParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

> **Note**
>
> There are five change types for creating an SSM patch baseline, for the various operating systems.

> **Important**
>
> At least one of **ApprovalRules** or **ApprovedPatches** is required.

If you create a patch baseline, it must have at least one approval rule and/or approved patch defined. An approval rule allows you to specify which classification (for example, SecurityUpdates) and severity (for example, Critical) patches will be installed. In your approval rules, you can define how many days after a patch is released it may be installed. A patch specified in the approved patches list will be installed irrespective of whether it is matched by an approval rule. Finally, items in the rejected patches list will exclude those patches from being installed, even if they match an approval rule and/or approved patch. For more information, see About predefined and custom patch baselines.

To create an SSM patch window, see Create SSM Patch Window.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-18fzkt86jmw1s.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SSM Patch Baseline | Create (Amazon Linux)

Create an AWS Systems Manager (SSM) patch baseline to define which patches are approved for installation on your instances for Amazon Linux OS. Specify existing instance "Patch Group" tag values for the patch baseline. The patch baseline is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch baseline | Create (Amazon Linux)

Change Type Details

Change type ID	ct-2taqdgegqthjr
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create for Amazon Linux

Creating an SSM Amazon Linux patch baseline with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

In the AWS Console, you can view the patch baselines you created at Systems Manager --> Patch Manager --> Patch Baselines.

Creating an SSM Amazon Linux patch baseline with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Amazon Linux:

```
aws amscm create-rfc --title Patch-Baseline-Create-AL-RFC --change-type-id
 ct-2taqdgegqthjr --change-type-version 1.0 --execution-parameters '{"ApprovalRules":
 [{"ApproveAfterDays": 7, "Classification": ["Security"], "Severity": ["All"]}],
 "OperatingSystem": "Amazon Linux", "Name": "TestBaselineAmazonLinux",
 "PatchGroupTagValues": ["MyAmazonLinuxPatchGroup"]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateALPatchBaselineParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2taqdgegqthjr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateALPatchBaselineParams.json
   ```

2. Modify and save the CreateALPatchBaselineParams file, make sure to modify these parameters to meet your specific needs.

   In this example, all critical security updates are approved for installation five days after release. Patches related to vulnerability ID CVE-2017-5754 are approved immediately even if they are not a critical severity. Finally, example-pkg-0.710.10-2.7.abcd.x86_64 will not be installed, even if it matches an approval rule.

   ```
   {
   "ApprovalRules":[{
       "ApproveAfterDays": 5,
       "Severity": [
           "Critical"
           ],
       "Classification": [
           "Security"
           ]
       }],
   "ApprovedPatches":["CVE-2017-5754"],
   "Description": "Patch Baseline",
   ```

```
"Name": "PatchBaseline-Unit-test",
"OperatingSystem": "Amazon Linux",
"PatchGroupTagValues": [
    "test1"
    ],
"RejectedPatches":["example-pkg-0.710.10-2.7.abcd.x86_64"],
"Tags": [
    {
    "Key":"patchGroupAL",
    "Value":"test1"
    }
]
```

3. Output the RFC template to a file in your current folder; this example names it CreateALPatchBaselineRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateALPatchBaselineRfc.json
```

4. Modify and save the CreateALPatchBaselineRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2taqdgegqthjr",
"Title":                "Patch-Baseline-Create-AL-RFC"
}
```

5. Create the RFC, specifying the CreateALPatchBaselineRfc file and the CreateALPatchBaselineParams file:

```
aws amscm create-rfc --cli-input-json file://CreateALPatchBaselineRfc.json --
execution-parameters file://CreateALPatchBaselineParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

> **Note**
>
> There are five change types for creating an SSM patch baseline, for the various operating systems.

> **Important**
>
> At least one of **ApprovalRules** or **ApprovedPatches** is required.

If you create a patch baseline, it must have at least one approval rule and/or approved patch defined. An approval rule allows you to specify which classification (for example, SecurityUpdates) and severity (for example, Critical) patches will be installed. In your approval rules, you can define how many days after a patch is released it may be installed. A patch specified in the approved patches list will be installed irrespective of whether it is matched by an approval rule. Finally, items in the rejected patches list will exclude those patches from being installed, even if they match an approval rule and/or approved patch. For more information, see About predefined and custom patch baselines.

To create an SSM patch window, see Create SSM Patch Window.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2taqdgegqthjr.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SSM Patch Baseline | Create (CentOS)

Create an AWS Systems Manager (SSM) patch baseline to define which patches are approved for installation on your instances for CentOS. Specify existing instance "Patch Group" tag values for the patch baseline. The patch baseline is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch baseline | Create (CentOS)

Change Type Details

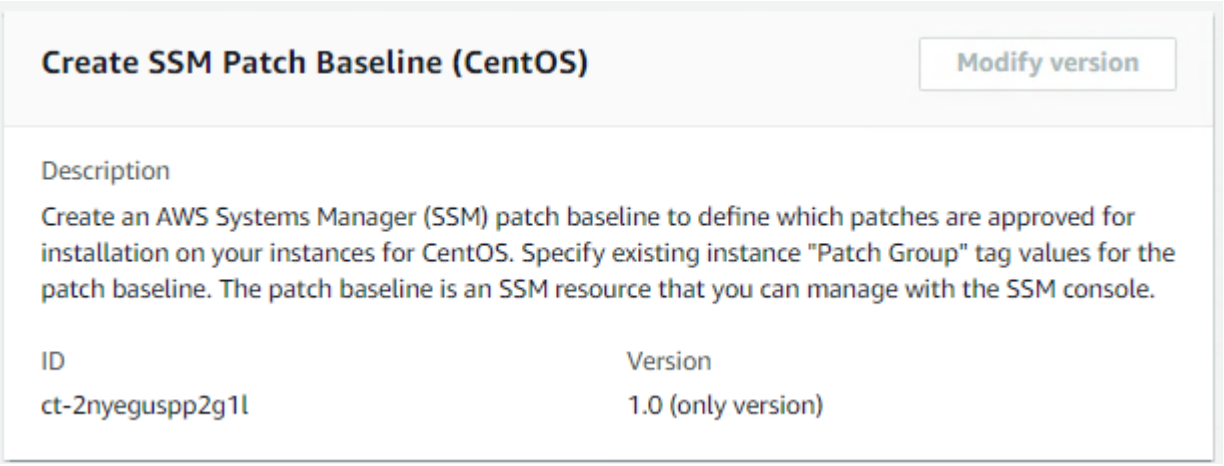
Change type ID	ct-2nyeguspp2g1l
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create for CentOS

Creating an SSM CentOS patch baseline with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

In the AWS Console, you can view the patch baselines you created at Systems Manager --> Patch Manager --> Patch Baselines.

Creating an SSM CentOS patch baseline with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

CentOS:

```
aws amscm create-rfc --title Patch-Baseline-Create-Centos-RFC --change-type-id
 ct-2nyeguspp2g1l --change-type-version 1.0 --execution-parameters '{"ApprovalRules":
 [{"ApproveAfterDays": 7, "Classification": ["All"], "Severity": ["All"]}],
 "OperatingSystem": "CentOS", "Name": "TestBaselineCentOS", "PatchGroupTagValues":
 ["MyCentOSPatchGroup"]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateCentosPatchBaselineParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-2taqdgegqthjr"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateCentosPatchBaselineParams.json
```

2. Modify and save the CreateCentosPatchBaselineParams file. See examples below; make sure to modify these parameters to meet your specific needs.

   In this example, all updates are approved for installation five days after release. The package **example-pkg-0.710.10-2.7.abcd.x86_64** will not be installed.

```
{
"ApprovalRules":[{
    "ApproveAfterDays": 5,
    "Severity": [
        "All"
        ],
    "Classification": [
        "All"
        ]
    }],
"Description": "Patch Baseline",
"Name": "PatchBaseline-Unit-test",
"OperatingSystem": "CentOS",
"PatchGroupTagValues": [
    "test1"
    ],
"RejectedPatches":["example-pkg-0.710.10-2.7.abcd.x86_64"],
"Tags": [
    {
    "Key":"patchGroupCent",
    "Value":"test1"
    }
]
```

3. Output the RFC template to a file in your current folder; this example names it CreateCentosPatchBaselineRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateCentosPatchBaselineRfc.json
```

4. Modify and save the CreateCentosPatchBaselineRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2nyeguspp2g1l",
"Title":                "Patch-Baseline-Create-Centos-RFC"
}
```

5. Create the RFC, specifying the CreateCentosPatchBaselineRfc file and the CreateCentosPatchBaselineParams file:

```
aws amscm create-rfc --cli-input-json file://CreateCentosPatchBaselineRfc.json --
execution-parameters file://CreateCentosPatchBaselineParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

> **Note**
>
> There are five change types for creating an SSM patch baseline, for the various operating systems.

> **Note**
>
> Because CentOS default repos do not provide update notice metadata, all updates are categorized as non-security, with no classification or severity info. As a result, you must specify **Classification:All** and **Severity:All** for any CentOS patch baseline. If you do not allow non-security updates in your CentOS baseline, no updates of any type will be installed from default repos. For more detail, see How security patches are selected.

> **Important**
>
> At least one of **ApprovalRules** or **ApprovedPatches** is required.

If you create a patch baseline, it must have at least one approval rule and/or approved patch defined. An approval rule allows you to specify which classification (for example, SecurityUpdates) and severity (for example, Critical) patches will be installed. In your approval rules, you can define how many days after a patch is released it may be installed. A patch specified in the approved patches list will be installed irrespective of whether it is matched by an approval rule. Finally, items in the rejected patches list will exclude those patches from being installed, even if they match an approval rule and/or approved patch. For more information, see About predefined and custom patch baselines.

To create an SSM patch window, see Create SSM Patch Window.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2nyeguspp2g1l.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SSM Patch Baseline | Create (Red Hat)

Create an AWS Systems Manager (SSM) patch baseline to define which patches are approved for installation on your instances for RHEL OS. Specify existing instance "Patch Group" tag values for the patch baseline. The patch baseline is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch baseline | Create (Red Hat)

Change Type Details

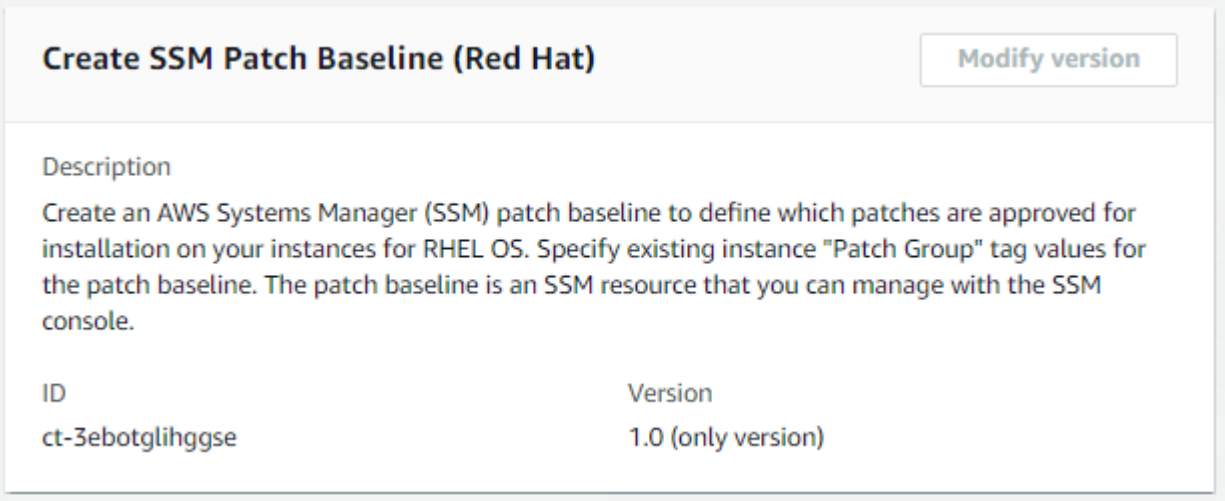
Change type ID	ct-3ebotglihggse
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create for RHEL

Creating an SSM RHEL patch baseline with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

In the AWS Console, you can view the patch baselines you created at Systems Manager --> Patch Manager --> Patch Baselines.

Creating an SSM RHEL patch baseline with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title Patch-Baseline-Create-Rhel-RFC --change-type-id
 ct-3ebotglihggse --change-type-version 1.0 --execution-parameters '{"ApprovalRules":
 [{"ApproveAfterDays": 7, "Classification": ["Security"], "Severity": ["All"]}],
 "OperatingSystem": "Red Hat Enterprise Linux", "Name": "TestBaselineRHEL",
 "PatchGroupTagValues": ["MyRHELPatchGroup"]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateRhelPatchBaselineParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2taqdgegqthjr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateRhelPatchBaselineParams.json
   ```

2. Modify and save the CreateRhelPatchBaselineParams file. See examples below; make sure to modify these parameters to meet your specific needs.

In this example, all critical security updates are approved for installation five days after release. Patches included in Red Hat Security Advisory RHSA-2018:0151 are approved immediately even if they are not a critical severity. Finally, example-pkg-0.710.10-2.7.abcd.x86_64 will not be installed, even if it matches an approval rule.

```
{
"ApprovalRules":[{
    "ApproveAfterDays": 5,
    "Severity": [
        "Critical"
        ],
    "Classification": [
        "Security"
        ]
    }],
"ApprovedPatches":["RHSA-2018:0151"],
"Description": "Patch Baseline",
"Name": "PatchBaseline-Unit-test",
"OperatingSystem": "Red Hat Enterprise Linux",
"PatchGroupTagValues": [
    "test1"
    ],
"RejectedPatches":["example-pkg-0.710.10-2.7.abcd.x86_64"],
"Tags": [
    {
    "Key":"patchGroupRhel",
    "Value":"test1"
    }
]
```

3. Output the RFC template to a file in your current folder; this example names it CreateRhelPatchBaselineRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateRhelPatchBaselineRfc.json
```

4. Modify and save the CreateRhelPatchBaselineRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3ebotglihggse",
```

```
"Title":                "Patch-Baseline-Create-Rhel-RFC"
}
```

5. Create the RFC, specifying the CreateRhelPatchBaselineRfc file and the CreateRhelPatchBaselineParams file:

```
aws amscm create-rfc --cli-input-json file://CreateRhelPatchBaselineRfc.json --
execution-parameters file://CreateRhelPatchBaselineParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

> **Note**
>
> There are five change types for creating an SSM patch baseline, for the various operating systems.

> **Important**
>
> At least one of **ApprovalRules** or **ApprovedPatches** is required.

If you create a patch baseline, it must have at least one approval rule and/or approved patch defined. An approval rule allows you to specify which classification (for example, SecurityUpdates) and severity (for example, Critical) patches will be installed. In your approval rules, you can define how many days after a patch is released it may be installed. A patch specified in the approved patches list will be installed irrespective of whether it is matched by an approval rule. Finally, items in the rejected patches list will exclude those patches from being installed, even if they match an approval rule and/or approved patch. For more information, see About predefined and custom patch baselines.

To create an SSM patch window, see Create SSM Patch Window.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3ebotglihggse.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SSM Patch Baseline | Create (Windows)

Create an AWS Systems Manager (SSM) patch baseline to define which patches are approved for installation on your instances for Windows OS. Specify existing instance "Patch Group" tag values for the patch baseline. The patch baseline is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch baseline | Create (Windows)

Change Type Details

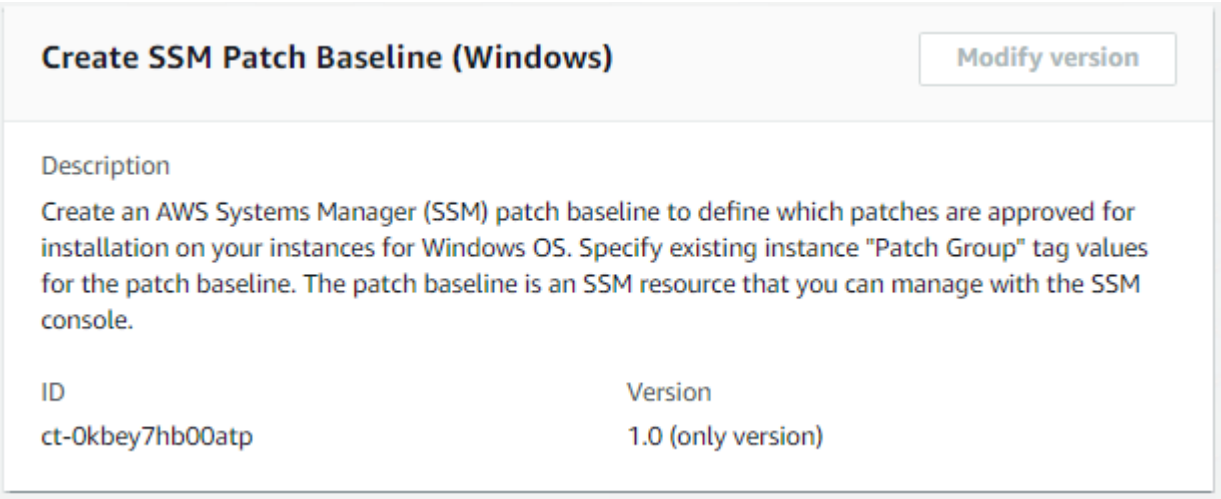
Change type ID	ct-0kbey7hb00atp
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create for Windows

Creating an SSM Windows patch baseline with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

In the AWS Console, you can view the patch baselines you created at Systems Manager --> Patch Manager --> Patch Baselines.

Creating an SSM Windows patch baseline with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title Patch-Baseline-Create-Win-RFC --change-type-id
 ct-0kbey7hb00atp --change-type-version 1.0 --execution-parameters '{"ApprovalRules":
 [{"ApproveAfterDays": 7, "Classification": ["SecurityUpdates"], "Severity": ["All"]}],
 "OperatingSystem": "Windows", "Name": "TestBaselineWindows", "PatchGroupTagValues":
 ["MyWindowsPatchGroup"]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateWinPatchBaselineParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2taqdgegqthjr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateWinPatchBaselineParams.json
   ```

2. Modify and save the CreateWinPatchBaselineParams file. See examples below; make sure to modify these parameters to meet your specific needs.

   In this example, all critical security updates are approved for installation five days after release. KB2032276 is approved immediately even if it is not a critical severity. Finally, KB2124261 will not be installed, even if it matches an approval rule.

   ```
   {
   "ApprovalRules":[{
       "ApproveAfterDays": 5,
       "Severity": [
           "Critical"
           ],
       "Classification": [
           "SecurityUpdates"
           ]
   ```

```
    }],
"ApprovedPatches":["KB2032276"],
"Description": "Patch Baseline",
"Name": "PatchBaseline-Unit-test",
"OperatingSystem": "Windows",
"PatchGroupTagValues": [
    "test1"
    ],
"RejectedPatches":["KB2124261"],
"Tags": [
    {
    "Key":"patchGroupWin",
    "Value":"test1"
    }
]
```

3. Output the RFC template to a file in your current folder; this example names it CreateWinPatchBaselineRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateWinPatchBaselineRfc.json
```

4. Modify and save the CreateWinPatchBaselineRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0kbey7hb00atp",
"Title":                "Patch-Baseline-Create-Win-RFC"
}
```

5. Create the RFC, specifying the CreateWinPatchBaselineRfc file and the CreateWinPatchBaselineParams file:

```
aws amscm create-rfc --cli-input-json file://CreateWinPatchBaselineRfc.json --
execution-parameters file://CreateWinPatchBaselineParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

> **Note**
>
> There are five change types for creating an SSM patch baseline, for the various operating systems.

> **Important**
>
> At least one of **ApprovalRules** or **ApprovedPatches** is required.

If you create a patch baseline, it must have at least one approval rule and/or approved patch defined. An approval rule allows you to specify which classification (for example, SecurityUpdates) and severity (for example, Critical) patches will be installed. In your approval rules, you can define how many days after a patch is released it may be installed. A patch specified in the approved patches list will be installed irrespective of whether it is matched by an approval rule. Finally, items in the rejected patches list will exclude those patches from being installed, even if they match an approval rule and/or approved patch. For more information, see About predefined and custom patch baselines.

To create an SSM patch window, see Create SSM Patch Window.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0kbey7hb00atp.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SSM Patch Window | Create

Create an AWS Systems Manager (SSM) patch window for patching to take place on instances with the specified PatchGroup. The patch window is an SSM resource that you can manage with the SSM console.

Full classification: Deployment | Patching | SSM patch window | Create

Change Type Details

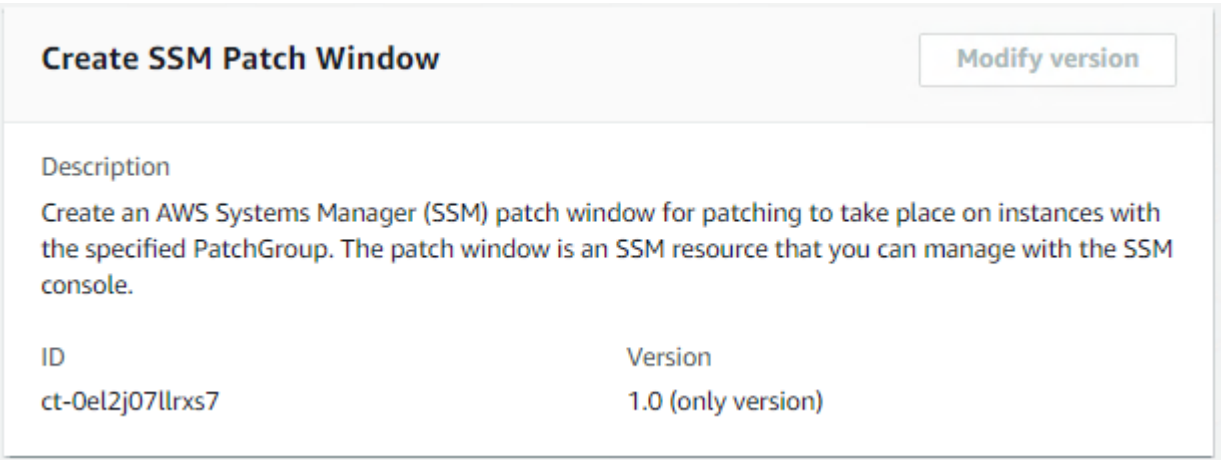
Change type ID	ct-0el2j07llrxs7
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create SSM Patch Window

Creating an SSM patch window with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an SSM patch window with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id ID` command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title my-test-patchwindow --changetype-id ct-0el2j07llrxs7 --
change-type-version 1.0 --execution-parameters '{"Cutoff":2, "Description":"Test",
 "Duration":24, "MaxConcurrency":"10", "MaxErrors":"12", "NotificationEmails":
["test@supertest.com"], "PatchGroup":"test-patch-group", "Schedule":"cron(0 3 ? * 6L
 *)", "ScheduleTimeZone": "Africa/Harare"}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreatePatchWindowParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0el2j07llrxs7"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreatePatchWindowParams.json
   ```

2. Modify and save the CreatePatchWindowParams file.

```
{
"Cutoff": 23,
"Description": "Required param given test",
"Duration": 24,
"EndDate": "2008-09-15T15:53:00Z",
"MaxConcurrency": "10",
"MaxErrors": "12",
"Name": "Test1",
"NotificationEmails": ["email@example.com"],
"PatchGroup": "Prod",
"Schedule": "cron(0 3 ? * 6L *)",
"ScheduleTimeZone": "Africa/Harare",
"ScheduleOffset": "0",
"StartDate": "2008-09-15T15:53:00Z"
}
```

3. Output the RFC template to a file in your current folder; this example names it CreatePatchWindowRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreatePatchWindowRfc.json
```

4. Modify and save the CreatePatchWindowRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0el2j07llrxs7",
"Title":                "Patch-Window-Create-RFC"
}
```

5. Create the RFC, specifying the CreatePatchWindowRfc file and the CreatePatchWindowParams file:

```
aws amscm create-rfc --cli-input-json file://CreatePatchWindowRfc.json --execution-
parameters file://CreatePatchWindowParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

- To learn more about AWS SSM patch windows, see "Maintenance Window" at Patching your Windows EC2 instances using AWS Systems Manager Patch Manager.
- To create an SSM patch baseline, see SSM Patch Window | Create.

  To update a custom Maintenance Window, see Updating an SSM Patch Window.

  To delete a custom Maintenance Window, see Delete stack.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0el2j07llrxs7.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Cutoff": 23,
  "Description": "Required param given test",
  "Duration": 24,
  "EndDate": "2008-09-15T15:53:00Z",
  "MaxConcurrency": "10",
  "MaxErrors": "12",
  "Name": "Test1",
  "NotificationEmails": ["email1@example.com"],
  "PatchGroup": "Prod",
  "Schedule": "cron(0 0 0 ? * 3#2 *)",
  "ScheduleOffset": 1,
  "ScheduleTimeZone": "Africa/Harare",
  "StartDate": "2008-09-15T15:53:00Z"
}
```

Standalone Resources Subcategory

Change Type Items and Operations in the Standalone Resources Subcategory

EC2 Instance | Create For WIGS (Review Required)

Create an Amazon Elastic Compute Cloud (EC2) instance for use with Workload Ingest (WIGS) change type (ct-257p9zjk14ija). For information, see AMS documentation on WIGS in the AMS Application Developer's Guide.

Full classification: Deployment | Standalone resources | EC2 instance | Create for WIGS (review required)

Change Type Details

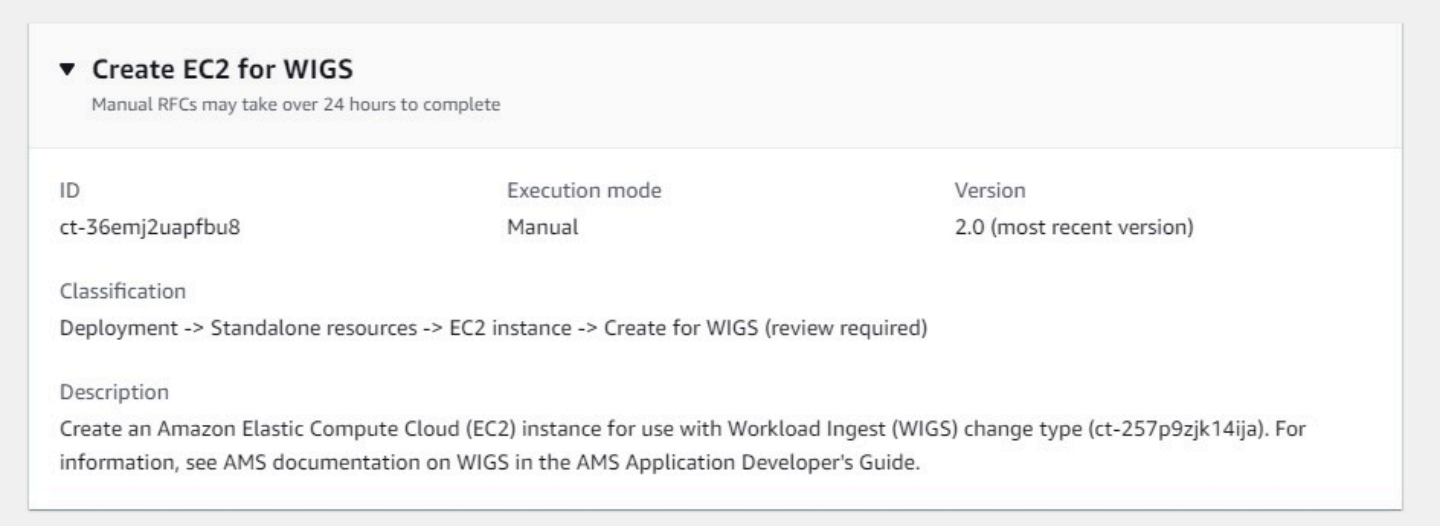
Change type ID	ct-36emj2uapfbu8
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create for WIGS (review required)

Creating an instance for WIGS with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an instance for WIGS with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-36emj2uapfbu8" --change-type-version "2.0"
 --title "Create Pre-Ingestion Instance" --execution-parameters "{\"InstanceVpcId
\": \"vpc-1234567890abcdef0\",\"InstanceAmiId\": \"ami-1234567890abcdef0\",
\"InstanceEBSOptimized\": false,\"InstanceRootVolumeSize\": 60,\"InstanceNameTagValue
\": \"temp-wigs\",\"InstanceType\": \"t3.large\",\"InstanceSubnetId\":
 \"subnet-0bb1c79de3EXAMPLE\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it CreateEc2PreIngestParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-36emj2uapfbu8"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateEc2PreIngestParams.json
   ```

2. Modify and save the CreateEc2PreIngestParams file. For example, you can replace the contents with something like this:

   ```
   {
       "InstanceVpcId": "vpc-1234567890abcdef0",
       "InstanceAmiId": "ami-1234567890abcdef0",
       "InstanceEBSOptimized": false,
       "InstanceRootVolumeSize": 60,
       "InstanceSubnetId": "subnet-1234567890abcdef0",
       "InstanceType": "t3.large",
       "InstanceNameTagValue": "temp-wigs",
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it CreateEc2PreIngestRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateEc2PreIngestRfc.json
   ```

4. Modify and save the CreateEc2PreIngestRfc.json file. For example, you can replace the contents with something like this:.

```
{
    "ChangeTypeVersion": "2.0",
    "ChangeTypeId": "ct-36emj2uapfbu8",
    "Title": "Create Pre-Ingestion Instance"
}
```

5. Create the RFC, specifying the CreateEc2PreIngestRfc file and the CreateEc2PreIngestParams file:

```
aws amscm create-rfc --cli-input-json file://CreateEc2PreIngestRfc.json  --
execution-parameters file://CreateEc2PreIngestParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-36emj2uapfbu8.

Example: Required Parameters

```
{
    "InstanceVpcId": "vpc-1234567890abcdef0",
    "InstanceAmiId": "ami-1234567890abcdef0",
    "InstanceSubnetId": "subnet-1234567890abcdef0"
}
```

Example: All Parameters

```
{
    "InstanceVpcId": "vpc-1234567890abcdef0",
    "InstanceAmiId": "ami-1234567890abcdef0",
    "InstanceEBSOptimized": false,
    "InstanceRootVolumeSize": 60,
    "InstanceSubnetId": "subnet-1234567890abcdef0",
    "InstanceType": "t3.large",
    "InstanceNameTagValue": "temp-wigs",
    "Priority": "Medium"
}
```

Standard Stacks Subcategory

Change Type Items and Operations in the Standard Stacks Subcategory

- High Availability One-Tier Stack | Create
- High Availability One-Tier Stack | Create (With ELB)
- High Availability Two-Tier Stack | Create

High Availability One-Tier Stack | Create

Use to create an Application Load Balancer and an Auto Scaling Group.

Full classification: Deployment | Standard stacks | High availability one-tier stack | Create

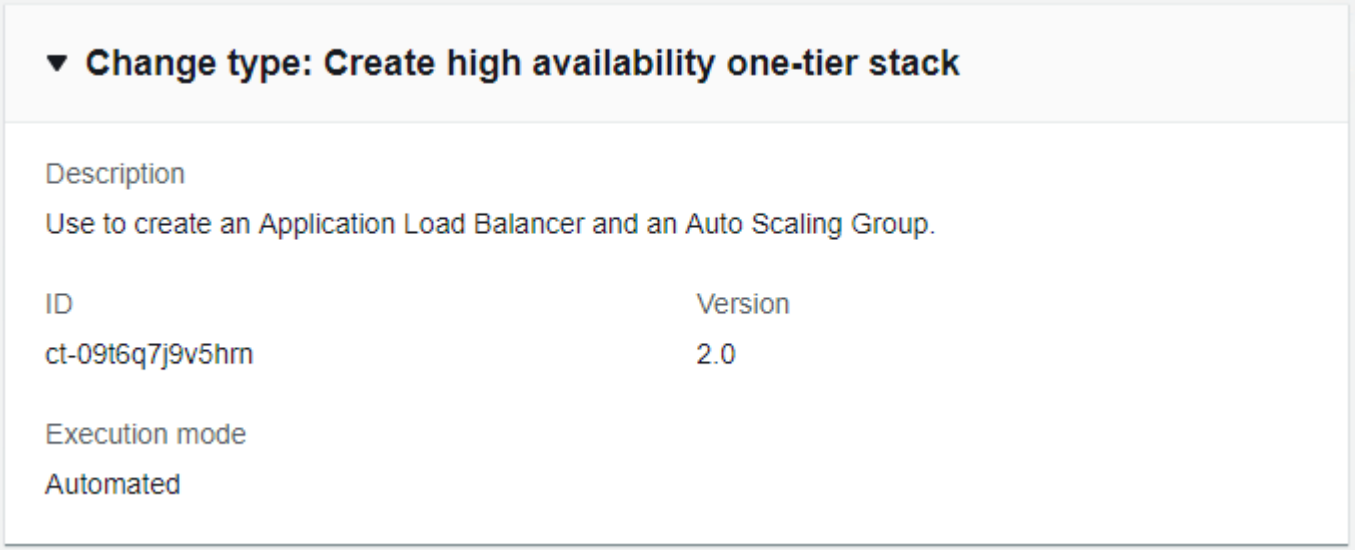
Change Type Details

Change type ID	ct-09t6q7j9v5hrn
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

High availability one-tier stacks: Creating

Creating a high availability one-tier stack with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a high availability one-tier stack with the CLI

How it works:

1. Use the Template Create method (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CreateOnetierStackParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-09t6q7j9v5hrn"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateOnetierStackParams.json
```

2. Modify the schema, replacing the *variables* as appropriate.

```
{
    "Description":          "HA-One-Tier-Stack",
    "Name":                 "One-Tier-Stack",
    "TimeoutInMinutes":     "360",
    "VpcId":                "VPC_ID",
    "ApplicationLoadBalancer": {
        "SubnetIds": [
                  "SUBNET_ID",
                  "SUBNET_ID"
        ]
    },
    "AutoScalingGroup": {
        "AmiId": "AMI-ID"
        "SubnetIds": [
                 "SUBNET_ID",
                 "SUBNET_ID"
        ]
    }
}
```

3. Output the CreateRfc JSON template to a file in your current folder; example names it CreateOnetierStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateOnetierStackRfc.json
```

4. Modify the RFC template as appropriate and save it. Reset the start and end times for a scheduled RFC, or leave off for an ASAP RFC.

```
{
"ChangeTypeVersion":    2.0",
"ChangeTypeId":         "ct-09t6q7j9v5hrn",
"Title":                "HA-One-Tier-RFC",
"RequestedStartTime":   "2019-04-28T22:45:00Z",
```

```
"RequestedEndTime":     "2019-04-28T22:45:00Z"
}
```

5. Create the RFC, specifying the CreateOnetierStackRfc.json file and the CreateOnetierStackParams.json execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateOnetierStackRfc.json --
execution-parameters file://CreateOnetierStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a large provisioning of resources, especially if you add UserData. The load balancer Amazon resource name (ARN) can be found through the Load Balancer page of the EC2 console by searching with the load balancer stack ID returned in the RFC execution output.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-09t6q7j9v5hrn.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "TimeoutInMinutes": 360,
  "ApplicationLoadBalancer": {
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"]
  },
  "AutoScalingGroup": {
    "AmiId": "ami-01234567890abcdef",
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"]
  },
  "Description": "This stack contains an ALB and an ASG.",
  "Name": "High availability one-tier stack"
```

```
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678",
  "TimeoutInMinutes": 360,
  "DatabaseStackId": "stack-0123456789abcdefg",
  "Description": "This stack contains an ALB and an ASG.",
  "Name": "High availability one-tier stack",
  "Tags": [
    {
      "Key": "Foo",
      "Value": "Bar"
    }
  ],
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567",
  "ApplicationLoadBalancer": {
    "HealthCheckHealthyThreshold": 2,
    "HealthCheckIntervalInSeconds": 10,
    "HealthCheckTargetPath": "/",
    "HealthCheckTargetPort": 80,
    "HealthCheckTargetProtocol": "HTTPS",
    "HealthCheckTimeoutSeconds": 5,
    "HealthCheckUnhealthyThreshold": 2,
    "InstancePort": 80,
    "InstanceProtocol": "HTTP",
    "LoadBalancerCookieExpirationPeriodInSeconds": 3600,
    "LoadBalancerPort": 80,
    "LoadBalancerAccessCIDRRange": "1.2.3.4/0",
    "LoadBalancerProtocol": "HTTP",
    "LoadBalancerSslPolicy": "ELBSecurityPolicy-2016-08",
    "Public": false,
    "SSLCertificateId": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
    "SubnetIds": ["subnet-a0b1c2d3", "subnet-e4f5g6h7"],
    "ValidHTTPCode": "200"
  },
  "AutoScalingGroup": {
    "AmiId": "ami-01234567",
    "CooldownInSeconds": 300,
    "DesiredCapacity": 1,
```

```
    "EBSOptimized": false,
    "HealthCheckGracePeriodInSeconds": 1800,
    "HealthCheckType": "EC2",
    "IAMInstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceDetailedMonitoring" : true,
    "InstanceRootVolumeIops" : 0,
    "InstanceRootVolumeName": "/dev/xvda",
    "InstanceRootVolumeSizeInGiB" : 20,
    "InstanceRootVolumeType" : "standard",
    "InstanceType": "m4.large",
    "MaxInstances": 1,
    "MinInstances": 1,
    "ScaleMetricName": "CPUUtilization",
    "ScaleDownPolicyCooldownInSeconds": 300,
    "ScaleDownPolicyEvaluationPeriods": 4,
    "ScaleDownPolicyPeriod": 60,
    "ScaleDownPolicyScalingAdjustment": -1,
    "ScaleDownPolicyStatistic": "Average",
    "ScaleDownPolicyThreshold": 35,
    "ScaleUpPolicyCooldownInSeconds": 300,
    "ScaleUpPolicyEvaluationPeriods": 2,
    "ScaleUpPolicyPeriod": 60,
    "ScaleUpPolicyScalingAdjustment": 1,
    "ScaleUpPolicyStatistic": "Average",
    "ScaleUpPolicyThreshold": 75,
    "SubnetIds": ["subnet-a0b1c2d3", "subnet-e4f5g6h7"],
    "UserData": ["#!/bin/bash","echo hello"]
  }
}
```

High Availability One-Tier Stack | Create (With ELB)

Create a stack with an Auto Scaling Group, and an Elastic Load Balancer (ELB) with up to two listeners, integrated with an existing security group that you specify.

Full classification: Deployment | Standard stacks | High availability one-tier stack | Create (with ELB)

Change Type Details

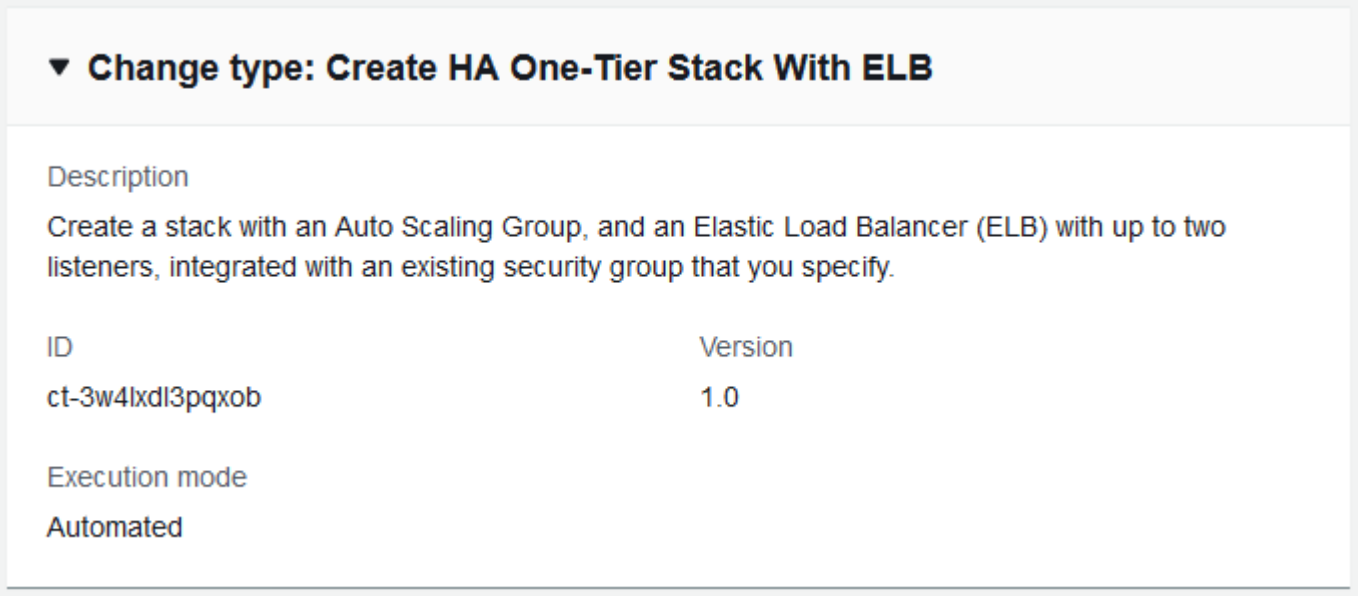
Change type ID	ct-3w4lxdl3pqxob

Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

High availability one-tier Stacks: Creating (with ELB)

Creating a high availability one-tier stack with an ELB with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a high availability one-tier stack with an ELB with the CLI

How it works:

1. Use the Template Create method (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-3w4lxdl3pqxob" --change-type-version "1.0" --title 'Test - HA' --description
 "Test Stack"  --execution-parameters "{\"Description\":\"DESCRIPTION\",\"VpcId
\":\"VPC_ID\",\"Name\":\"TestStack\",\"StackTemplateId\":\"stm-g7rc538l62r4c23nb
\",\"TimeoutInMinutes\":60,\"AutoScaling\":{\"AmiId\":\"AMI_ID\",\"SubnetIds\":
[\"SUBNET_ID\"]},\"LoadBalancer\":{\"SecurityGroups\":\"SG_ID\",\"SubnetIds\":
[\"SUBNET_ID\"]},\"Listener1\":{\"Port\":\"443\",\"Protocol\":\"HTTPS\",\"InstancePort
\":\"443\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CreateOnetierElbStackParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3w4lxdl3pqxob"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateOnetierElbStackParams.json
   ```

2. Modify the schema, replacing the *`variables`* as appropriate.

   ```
   {
     "Description" : "DESCRIPTION",
     "VpcId" : "VPC_ID",
     "Name" : "TestStack",
   ```

```
  "StackTemplateId" : "stm-g7rc538l62r4c23nb",
  "TimeoutInMinutes" : 60,
  "AutoScaling" : {
    "AmiId" : "AMI_ID",
    "SubnetIds": ["SUBNET_ID"]
  },
  "LoadBalancer" : {
    "SecurityGroups" : "SG_ID",
    "SubnetIds" : ["SUBNET_ID"]
  },
  "Listener1" : {
    "Port" : "443",
    "Protocol" : "HTTPS",
    "InstancePort" : "443"
  }
}
```

3. Output the CreateRfc JSON template to a file in your current folder; example names it CreateOnetierElbStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateOnetierElbStackRfc.json
```

4. Modify the RFC template as appropriate and save it. Reset the start and end times for a scheduled RFC, or leave off for an ASAP RFC.

```
{
"ChangeTypeVersion":    1.0",
"ChangeTypeId":         "ct-3w4lxdl3pqxob",
"Title":                "HA-One-Tier-ELB-RFC",
"RequestedStartTime":   "2019-04-28T22:45:00Z",
"RequestedEndTime":     "2019-04-28T22:45:00Z"
}
```

5. Create the RFC, specifying the CreateOnetierElbStackRfc.json file and the CreateOnetierElbStackParams.json execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CreateOnetierElbStackRfc.json --
execution-parameters file://CreateOnetierElbStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a large provisioning of resources, especially if you add UserData. The load balancer Amazon Resource Name (ARN) can be found through the Load Balancer page of the EC2 console by searching with the load balancer stack ID returned in the RFC execution output.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3w4lxdl3pqxob.

Example: Required Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678901234567",
  "Name" : "TestStack",
  "StackTemplateId" : "stm-g7rc538l62r4c23nb",
  "TimeoutInMinutes" : 60,
  "AutoScaling" : {
    "AmiId" : "ami-12345678901234567",
    "SubnetIds": ["subnet-12345678"]
  },
  "LoadBalancer" : {
    "SecurityGroups" : "sg-12345678901234567",
    "SubnetIds" : ["subnet-12345678901234567"]
  },
  "Listener1" : {
    "Port" : "443",
    "Protocol" : "HTTPS",
    "InstancePort" : "443"
  }
}
```

Example: All Parameters

```
{
  "Description" : "Test description",
  "VpcId" : "vpc-12345678",
```

```
  "Name" : "TestStack",
  "Tags" : [
    {
      "Key" : "foo",
      "Value" : "bar"
    }
  ],
  "StackTemplateId" : "stm-g7rc538l62r4c23nb",
  "TimeoutInMinutes" : 60,
  "AutoScaling" : {
    "AmiId" : "ami-12345678",
    "InstanceType" : "m4.large",
    "RootVolumeIops" : "100",
    "RootVolumeName" : "/dev/xvda",
    "RootVolumeSize" : 100,
    "RootVolumeType" : "gp2",
    "EBSOptimized" : "false",
    "MaxInstances" : "1",
    "MinInstances" : "2",
    "IAMInstanceProfile" : "customer-mc-ec2-instance-profile",
    "SubnetIds": ["subnet-12345678"],
    "UserData": ["touch /tmp/test.out"],
    "MaxBatchSize" : 1,
    "MinInstancesInService" : 1,
    "HealthCheckType" : "EC2",
    "HealthCheckGracePeriod" : "600",
    "DetailedMonitoring" : "true",
    "Cooldown" : "300",
    "ScaleMetricName" : "CPUUtilization",
    "ScaleUpPolicyCooldown" : "60",
    "ScaleUpPolicyEvaluationPeriods" : "2",
    "ScaleUpPolicyPeriod" : "60",
    "ScaleUpPolicyScalingAdjustment" : "2",
    "ScaleUpPolicyStatistic" : "Average",
    "ScaleUpPolicyThreshold" : "75",
    "ScaleDownPolicyCooldown" : "300",
    "ScaleDownPolicyEvaluationPeriods" : "4",
    "ScaleDownPolicyPeriod" : "60",
    "ScaleDownPolicyScalingAdjustment" : "-1",
    "ScaleDownPolicyStatistic" : "Average",
    "ScaleDownPolicyThreshold" : "35"
  },
  "LoadBalancer" : {
    "Name" : "testLoadBalancer",
```

```
    "Public" : "false",
    "SecurityGroups" : "sg-12345678",
    "SubnetIds" : ["subnet-12345678"],
    "AccessLogInterval" : "60",
    "ConnectionDrainingTimeout" : 60,
    "IdleTimeout" : 60,
    "CrossZone" : "true",
    "HealthCheckHealthyThreshold" : "2",
    "HealthCheckInterval" : "10",
    "HealthCheckTarget" : "TCP:80",
    "HealthCheckTimeout" : "5",
    "HealthCheckUnhealthyThreshold" : "10",
    "LBCookieExpirationPeriod" : "2",
    "LBCookieStickinessPolicyName" : "LBCOOKIE",
    "AppCookieName": "APPCookie",
    "AppCookiePolicyName": "AppCookiePolicy"
  },
  "Listener1" : {
    "InstancePort" : "80",
    "InstanceProtocol" : "HTTP",
    "Port" : "443",
    "Protocol" : "HTTPS",
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  },
  "Listener2" : {
    "InstancePort" : "8080",
    "InstanceProtocol" : "HTTP",
    "Port" : "8443",
    "Protocol" : "HTTPS",
    "SSLCertificateId" : "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

High Availability Two-Tier Stack | Create

Creates a stack consisting of an Auto Scaling group, an RDS DB instance, and a load balancer (ELB). Optionally allows for application deployment with CodeDeploy by also creating a CodeDeploy application and deployment group both named the value given for ApplicationName. All resource parameters can be configured.

Full classification: Deployment | Standard stacks | High availability two-tier stack | Create

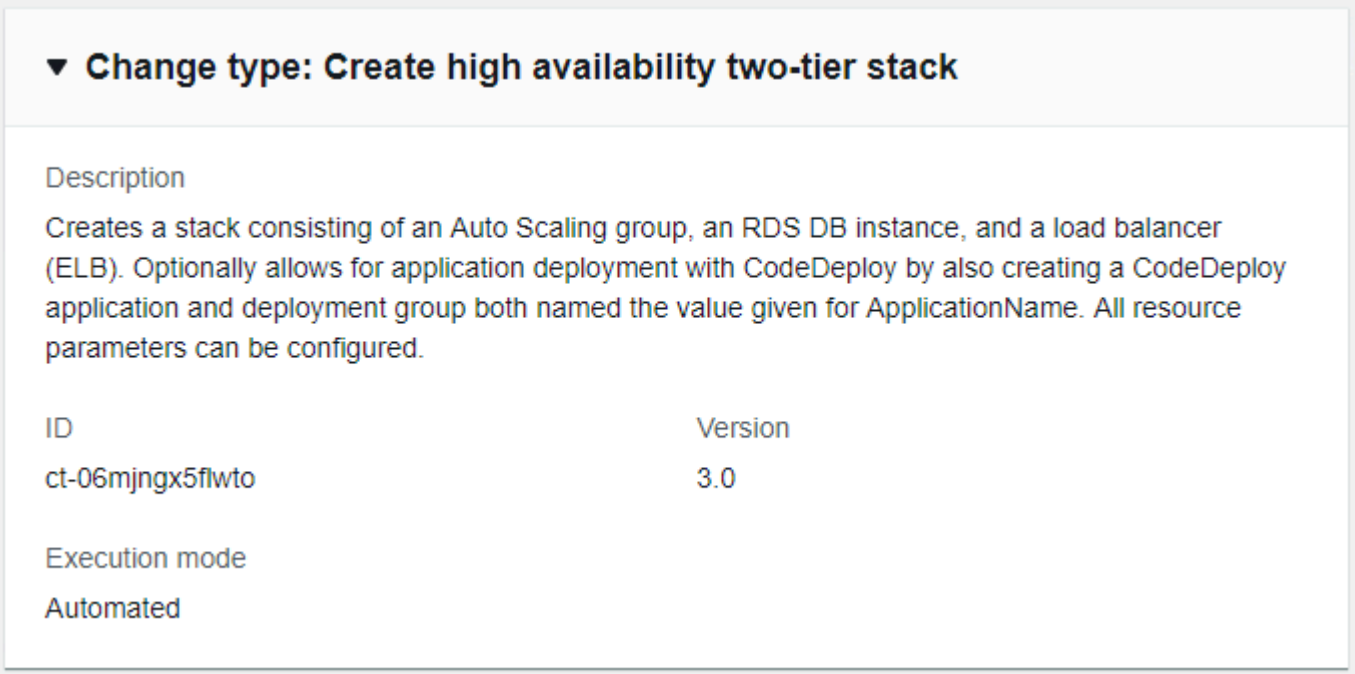
Change Type Details

Change type ID	ct-06mjngx5flwto
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create high availability two-tier stacks

Creating a high availability two-tier stack with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a high availability two-tier stack with the CLI

How it works:

1. Use the Template Create method (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it Create2tierStackParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-06mjngx5flwto"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    Create2tierStackParams.json
   ```

2. Modify the schema, replacing the *`variables`* as appropriate.

   ```
   {
     "Description":        "HA two tier stack",
     "Name":               "Two-Tier-Stack",
     "TimeoutInMinutes":   360,
     "VpcId":              "VPC-ID",
     "AutoScalingGroup": {
       "AmiId":            "AMI-ID",
       "SubnetIds": [
                   "Subnet-ID",
                   "Subnet-ID"
       ]
     },
     "Database": {
       "DBName":           "DB_Name",
       "DBEngine":         "postgres",
   ```

```
    "EngineVersion":    "9.6.3",
    "LicenseModel":     "postgresql-license",
    "MasterUsername":   "masterusername",
    "SubnetIds": [
                "Subnet-ID",
                "Subnet-ID"
    ]
  },
  "LoadBalancer": {
    "SubnetIds": [
                "Subnet-ID",
                "Subnet-ID"
    ]
  }
}
```

3. Output the CreateRfc JSON template to a file in your current folder; example names it Create2tierStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > Create2tierStackRfc.json
```

4. Modify the RFC template as appropriate and save it. Reset the start and end times for a scheduled RFC, or leave off for an ASAP RFC.

```
{
"ChangeTypeVersion":    3.0",
"ChangeTypeId":         "ct-06mjngx5flwto",
"Title":                "HA-2-Tier-RFC",
"RequestedStartTime":   "2019-04-28T22:45:00Z",
"RequestedEndTime":     "2019-04-28T22:45:00Z"
}
```

5. Create the RFC, specifying the Create2tierStackRfc.json file and the Create2tierStackParams.json execution parameters file:

```
aws amscm create-rfc --cli-input-json file://Create2tierStackRfc.json --execution-
parameters file://Create2tierStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a large provisioning of resources, especially if you add UserData. The load balancer Amazon resource name (ARN) can be found through the Load Balancer page of the EC2 console by searching with the load balancer stack ID returned in the RFC execution output.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-06mjngx5flwto.

Example: Required Parameters

```
{
  "Description": "My stack",
  "TimeoutInMinutes": 60,
  "VpcId": "vpc-01234567890abcdef",
  "Name": "MyStack",
  "AutoScalingGroup": {
    "AmiId" : "ami-01234567890abcdef",
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"]
  },
  "LoadBalancer": {
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"]
  },
  "Database": {
    "DBName": "main",
    "DBEngine": "MySQL",
    "EngineVersion": "4.5.6",
    "LicenseModel": "general-public-license",
    "MasterUsername": "admin",
    "MasterUserPassword": "adminpass",
    "SubnetIds": ["subnet-01234567890abcdef", "subnet-01234567891abcdef"]
  }
}
```

Example: All Parameters

```
{
```

```
  "Description": "My stack",
  "VpcId": "vpc-12345678",
  "TimeoutInMinutes": 60,
  "Name": "MyStack",
  "Tags": [
    {
      "Key": "Foo",
      "Value": "Bar"
    }
  ],
  "AutoScalingGroup": {
    "AmiId": "ami-12341234",
    "Cooldown": 120,
    "DesiredCapacity": 1,
    "EBSOptimized": false,
    "HealthCheckGracePeriod": 600,
    "IAMInstanceProfile": "foo",
    "InstanceDetailedMonitoring": true,
    "InstanceRootVolumeIops": 0,
    "InstanceRootVolumeName": "/dev/xvda",
    "InstanceRootVolumeSize": 30,
    "InstanceRootVolumeType": "gp2",
    "InstanceType": "m3.medium",
    "MaxInstances": 1,
    "MinInstances": 1,
    "ScaleDownPolicyCooldown": 300,
    "ScaleDownPolicyEvaluationPeriods": 4,
    "ScaleDownPolicyPeriod": 60,
    "ScaleDownPolicyScalingAdjustment": -1,
    "ScaleDownPolicyStatistic": "Average",
    "ScaleDownPolicyThreshold": 35,
    "ScaleMetricName": "CPUUtilization",
    "ScaleUpPolicyCooldown": 60,
    "ScaleUpPolicyEvaluationPeriods": 2,
    "ScaleUpPolicyPeriod": 60,
    "ScaleUpPolicyScalingAdjustment": 2,
    "ScaleUpPolicyStatistic": "Average",
    "ScaleUpPolicyThreshold": 75,
    "SubnetIds": ["subnet-a0b1c2d3", "subnet-e4f5g6h7"],
    "UserData": ["#!/bin/bash","echo hello"]
  },
  "LoadBalancer": {
    "SubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"],
    "HealthCheckInterval": 10,
```

```
    "HealthCheckTarget": "HTTP:80/index.html",
    "HealthCheckTimeout": 10,
    "Public": false,
    "AccessCIDRRange": "1.2.3.4/0"
  },
  "Database": {
    "AllocatedStorage": 100,
    "BackupRetentionPeriod": 7,
    "Backups": true,
    "DBEngine": "postgres",
    "DBName": "my_db",
    "EngineVersion": "9.5.2",
    "InstanceType": "db.m3.medium",
    "IOPS": 0,
    "LicenseModel": "postgresql-license",
    "MasterUsername": "myadminuser",
    "MasterUserPassword": "!#$%&')*+,-.0:;=>?AZ[\\^_`a{|~",
    "MultiAZ": false,
    "Port": 5432,
    "PreferredBackupWindow": "22:00-23:00",
    "PreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "StorageEncrypted": false,
    "StorageType": "gp2",
    "SubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"]
  },
  "Application": {
    "ApplicationName": "MyApplication",
    "DeploymentConfigName": "CodeDeployDefault.OneAtATime"
  },
  "EnforceIMDSv2": "optional"
}
```

Management Category

Change Type Subcategories in the Management Category

- Access Subcategory
- Advanced Stack Components Subcategory
- AMS Resource Scheduler Subcategory
- Applications Subcategory
- AWS Backup Subcategory

Access Subcategory

Change Type Items and Operations in the Access Subcategory

Stack Admin Access | Grant

Request admin access for one or more users for one or more stacks. The maximum access time is 12 hours.

Full classification: Management | Access | Stack admin access | Grant

Change Type Details

Change type ID	ct-1dmlg9g1l91h6

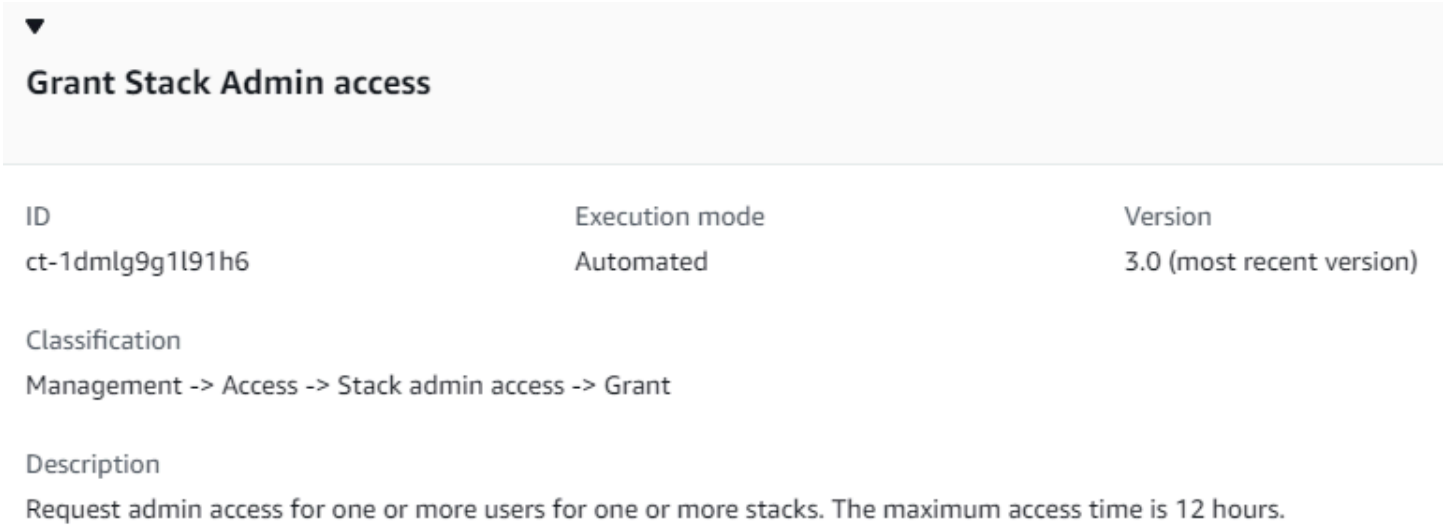
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Request administrative access

Requesting administrator access with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Requesting administrator access with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-1dmlg9g1l91h6" --change-type-
version "3.0" --title "Stack-Admin-Access-QC" --execution-parameters "{\"DomainFQDN
\":\"TEST.com\",\"StackIds\":[\"stack-01234567890abcdef\"],\"TimeRequestedInHours\":1,
\"Usernames\":[\"TEST\"],\"VpcId\":\"VPC_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it GrantAdminAccessParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1dmlg9g1l91h6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    GrantAdminAccessParams.json
   ```

   Modify and save the GrantAdminAccessParams file. For example, you can replace the contents with something like this:

   ```
   {
   "DomainFQDN":           "mycorpdomain.acme.com",
   "StackIds":             [STACK_ID, STACK_ID],
   "TimeRequestedInHours": 12,
   "Username":             ["USERNAME", "USERNAME"],
   "VpcId":                "VPC_ID"
   ```

```
}
```

Note that the `TimeRequestedInHours` option defaults to one hour. You can request up to twelve hours.

2. Output the RFC template to a file in your current folder; this example names it GrantAdminAccessRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > GrantAdminAccessRfc.json
```

3. Modify and save the GrantAdminAccessRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-1dmlg9g1l91h6",
"ChangeTypeVersion":     "3.0",
"Title":                 "Request-Admin-Access-to-EC2-RFC"
}
```

4. Create the RFC, specifying the GrantAdminAccessRfc file and the GrantAdminAcessParams file:

```
aws amscm create-rfc --cli-input-json file://GrantAdminAccessRfc.json --execution-
parameters file://GrantAdminAccessParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

To log in to the instance through a bastion, follow the next procedure, Instance access examples.

Tips

> **Note**
>
> You can submit an update to your access request before it expires. For information, see Stack Admin Access | Update.
> To log in to an instance that is part of an ASG, you request access to the ASG stack, which gives you access to all associated instances.

For an example about requesting ReadOnly access, see ReadOnly access: requesting.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1dmlg9g1l91h6.

Example: Required Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Example: All Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "TimeRequestedInHours": 1,
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Stack Admin Access | Update

Request admin access for one or more users for one or more stacks. The maximum access time is 12 hours.

Full classification: Management | Access | Stack admin access | Update

Change Type Details

Change type ID	ct-0ikpop8zqhkxg
Current version	3.0
Expected execution duration	60 minutes

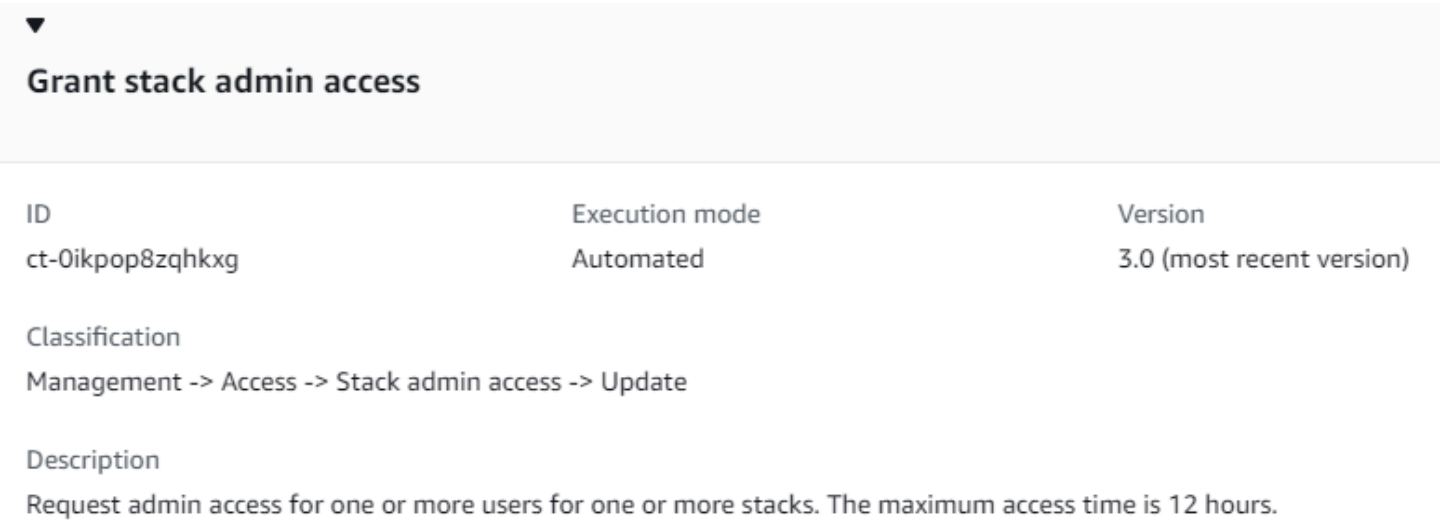
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update administrative access

Updating administrator access with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating administrator access with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-0ikpop8zqhkxg" --change-type-version "3.0" --title "Stack-Admin-Update-QC" --execution-parameters "{\"DomainFQDN\":\"TEST.com\",\"StackIds\":[\"stack-01234567890abcdef\"],\"TimeRequestedInHours\":1,\"Usernames\":[\"TEST\"],\"VpcId\":\"VPC_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateAdminAccessParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ikpop8zqhkxg"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateAdminAccessParams.json
   ```

   Modify and save the UpdateAdminAccessParams file. For example, you can replace the contents with something like this:

   ```
   {
   "DomainFQDN":           "mycorpdomain.acme.com",
   "StackIds":             [STACK_ID, STACK_ID],
   "TimeRequestedInHours": 12,
   "Usernames":            ["USERNAME", "USERNAME"],
   "VpcId":                "VPC_ID"
   }
   ```

   Note that the `TimeRequestedInHours` option defaults to one hour. You can request up to twelve hours.

2. Output the RFC template to a file in your current folder; this example names it UpdateAdminAccessRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateAdminAccessRfc.json
```

3. Modify and save the UpdateAdminAccessRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-0ikpop8zqhkxg",
"ChangeTypeVersion":     "3.0",
"Title":                 "Update-Admin-Access-to-EC2-RFC"
}
```

4. Create the RFC, specifying the UpdateAdminAccessRfc file and the UpdateAdminAcessParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateAdminAccessRfc.json --execution-
parameters file://UpdateAdminAccessParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

   To log in to the instance through a bastion, follow the next procedure, Instance access examples.

Tips

> **Note**
>
> To log in to an instance that is part of an EC2 Auto Scaling group (ASG), you request access to the ASG stack, which gives you access to all associated instances.

For a walkthrough on updating ReadOnly access, see ReadOnly Access: updating.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ikpop8zqhkxg.

Example: Required Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Example: All Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "TimeRequestedInHours": 1,
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Stack Read-Only Access | Grant

Request Read-Only access for one or more users for one or more stacks. The maximum access time is 12 hours.

Full classification: Management | Access | Stack read-only access | Grant

Change Type Details

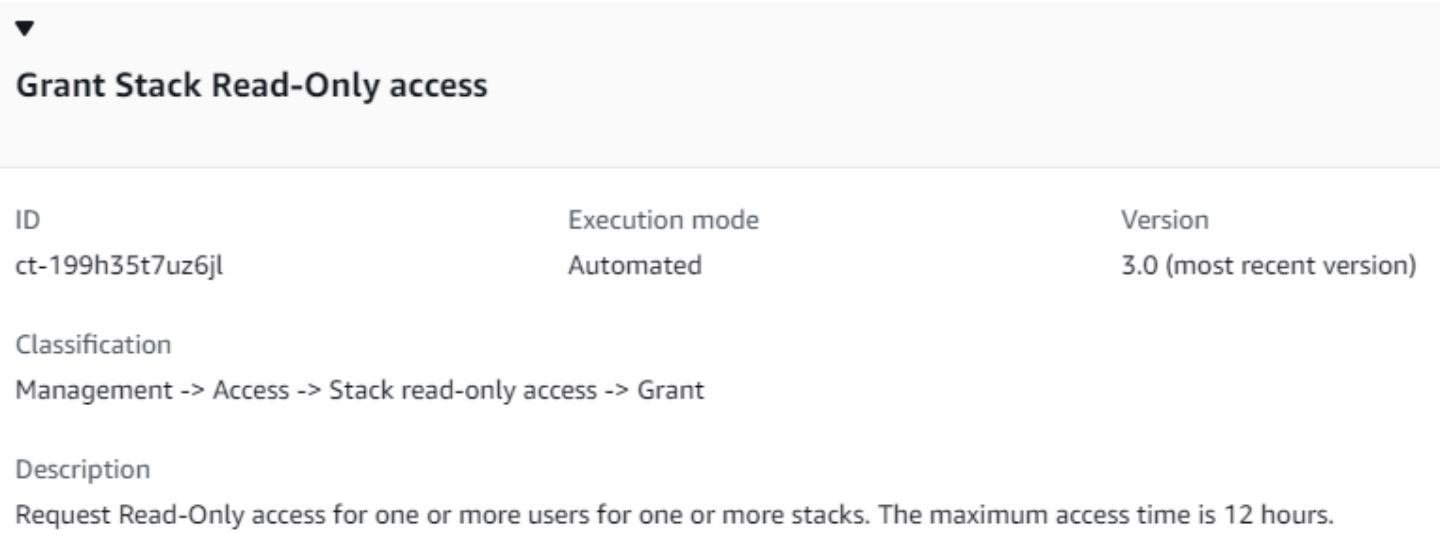
Change type ID	ct-199h35t7uz6jl
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Request ReadOnly access

Requesting ReadOnly access with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Requesting ReadOnly access with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-199h35t7uz6jl" --change-type-
version "3.0" --title "Stack-RO-Access-QC" --execution-parameters "{\"DomainFQDN\":
\"TEST.com\",\"StackIds\":[\"stack-01234567890abcdef\"],\"TimeRequestedInHours\":1,
\"Usernames\":[\"TEST\"],\"VpcId\":\"VPC_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it GrantReadOnlyAccessParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-199h35t7uz6jl"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    GrantReadOnlyAccessParams.json
   ```

   Modify and save the GrantReadOnlyAccessParams file. For example, you can replace the contents with something like this:

   ```
   {
   "DomainFQDN":           "mycorpdomain.acme.com",
   "StackIds":             [STACK_ID, STACK_ID],
   "TimeRequestedInHours": 12,
   "Usernames":             ["USERNAME", "USERNAME"],
   "VpcId":                "VPC_ID"
   }
   ```

   Note that the `TimeRequestedInHours` option defaults to one hour. You can request up to twelve hours.

2. Output the RFC template to a file in your current folder; this example names it GrantReadOnlyAccessRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > GrantReadOnlyAccessRfc.json
```

3. Modify and save the GrantReadOnlyAccessRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-199h35t7uz6jl",
"ChangeTypeVersion":     "3.0",
"Title":                 "Request-ReadOnly-Access-to-EC2-RFC"
}
```

4. Create the RFC, specifying the GrantReadOnlyAccessRfc file and the GrantReadOnlyAcessParams file:

```
aws amscm create-rfc --cli-input-json file://GrantReadOnlyAccessRfc.json --
execution-parameters file://GrantReadOnlyAccessParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

To log in to the instance through a bastion, follow the next procedure, Instance access examples.

Tips

> **Note**
>
> You can submit an update to your access request before it expires. For details, see Stack Read-Only Access | Update.
> To log into an instance that is part of an EC2 Auto Scaling group (ASG), you request access to the ASG stack, which gives you access to all associated instances.

For a walkthrough on requesting Admin access, see Admin Access: requesting.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-199h35t7uz6jl.

Example: Required Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Example: All Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "TimeRequestedInHours": 1,
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Stack Read-Only Access | Update

Request read only access for one or more users for one or more stacks. The maximum access time is 12 hours.

Full classification: Management | Access | Stack read-only access | Update

Change Type Details

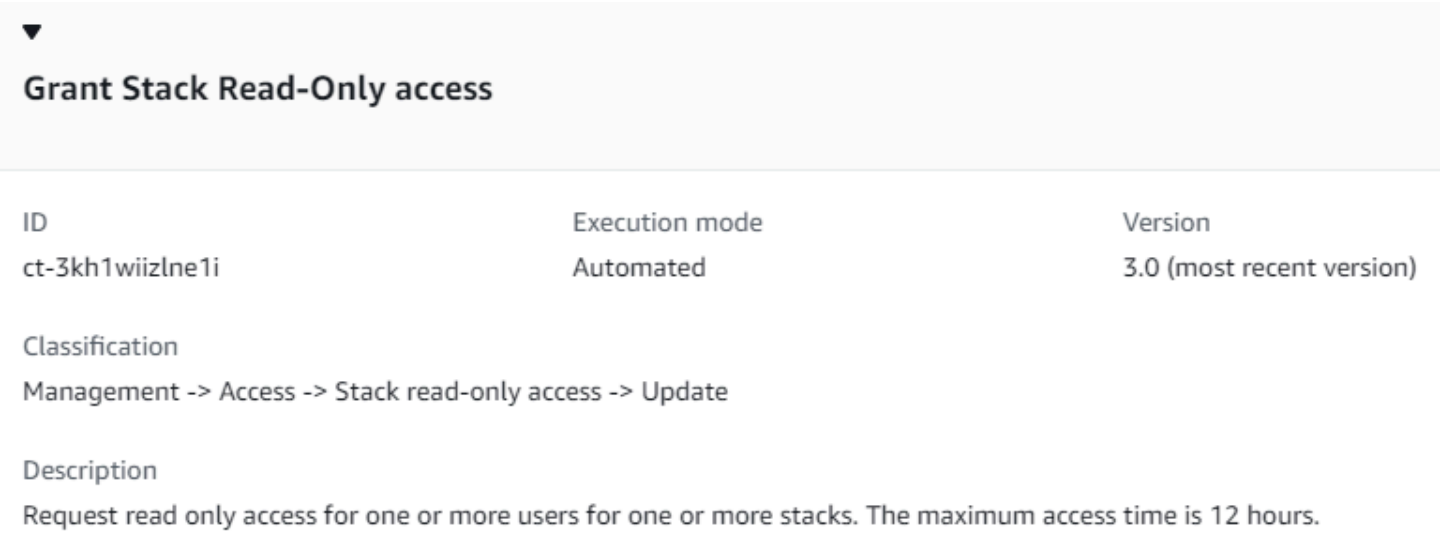
Change type ID	ct-3kh1wiizlne1i
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update ReadOnly access

Updating ReadOnly access with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating ReadOnly access with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3kh1wiizlne1i" --change-type-
version "3.0" --title "Stack-RO-Update-QC" --execution-parameters "{\"DomainFQDN\":
\"TEST.com\",\"StackIds\":[\"stack-01234567890abcdef\"],\"TimeRequestedInHours\":1,
\"Usernames\":[\"TEST\"],\"VpcId\":\"VPC_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateReadOnlyAccessParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3kh1wiizlne1i"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateReadOnlyAccessParams.json
   ```

   Modify and save the UpdateReadOnlyAccessParams.json file. For example, you can replace the contents with something like this:

   ```
   {
   "DomainFQDN":           "mycorpdomain.acme.com",
   "StackIds":             [STACK_ID, STACK_ID],
   "TimeRequestedInHours": 12,
   "Usernames":            ["USERNAME", "USERNAME"],
   "VpcId":                "VPC_ID"
   }
   ```

   Note that the `TimeRequestedInHours` option defaults to one hour. You may request up to twelve hours.

2. Output the RFC template to a file in your current folder; this example names it UpdateReadOnlyAccessRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateReadOnlyAccessRfc.json
```

3. Modify and save the UpdateReadOnlyAccessRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-3kh1wiizlne1i",
"ChangeTypeVersion":     "3.0",
"Title":                 "Update-ReadOnly-Access-to-EC2-RFC"
}
```

4. Create the RFC, specifying the UpdateReadOnlyAccessRfc file and the UpdateReadOnlyAcessParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateReadOnlyAccessRfc.json --
execution-parameters file://UpdateReadOnlyAccessParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

To log in to the instance through a bastion, follow the next procedure, Instance access examples.

Tips

For an example about updating Admin access, see Admin access: updating.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3kh1wiizlne1i.

Example: Required Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Example: All Parameters

```
{
  "DomainFQDN": "test.domain.com",
  "StackIds": ["stack-12345678901234567"],
  "TimeRequestedInHours": 1,
  "Usernames": ["AD_User_Name1"],
  "VpcId": "vpc-12345678"
}
```

Advanced Stack Components Subcategory

Change Type Items and Operations in the Advanced Stack Components Subcategory

- ACM | Delete Certificate
- AMI | Deregister
- AMI | Encrypt
- AMI | Share
- Application Load Balancer | Add Listener Certificate
- Application Load Balancer | Remove Listener Certificate
- Application Load Balancer | Update
- Auto Scaling Group | Update
- Bastions | Add CIDR Ingress (Review Required)
- Bastions | Update Instance or Session Counts (Review Required)
- Bastions | Update Instance Size (Review Required)
- Database Migration Service (DMS) | Start Replication Task
- Database Migration Service (DMS) | Stop Replication Task
- Directory Service | Accept Sharing
- DNS (Private) | Update
- DNS (Public) | Update
- EBS Snapshot | Archive
- EBS Snapshot | Delete
- EBS Snapshot | Share
- EBS Volume | Attach

- EBS Volume | Delete
- EBS Volume | Detach
- EBS Volume | Encrypt EBS By Default
- EBS Volume | Modify
- EBS Volume | Update
- EC2 Instance Stack | Associate Private IP Addresses (Review Required)
- EC2 Instance Stack | Change Hostname (Linux)
- EC2 Instance Stack | Change Hostname (Windows)
- EC2 Instance Stack | Change Time Zone
- EC2 Instance Stack | Enable Detailed Monitoring (Review Required)
- EC2 Instance Stack | Encrypt Instance Volumes
- EC2 Instance Stack | Gather Log4j Information
- EC2 Instance Stack | Reboot
- EC2 Instance Stack | Replace Instance Profile
- EC2 Instance Stack | Resize
- EC2 Instance Stack | Restore Volumes
- EC2 Instance Stack | Start
- EC2 Instance Stack | Stop
- EC2 Instance Stack | Update
- EC2 Instance Stack | Update (With Additional Volumes)
- EC2 Instance Stack | Update DeleteOnTermination (Review Required)
- EC2 Instance Stack | Update Instance Detailed Monitoring
- EC2 Instance Stack | Update Termination Protection
- Identity and Access Management (IAM) | Delete Account Alias
- Identity and Access Management (IAM) | Delete Entity or Policy (Read-Write Permissions)
- Identity and Access Management (IAM) | Delete Entity or Policy (Review Required)
- Identity and Access Management (IAM) | Delete or Deactivate Access Key
- Identity and Access Management (IAM) | Delete SAML Identity Provider
- Identity and Access Management (IAM) | Reset Service-Specific Credentials
- Identity and Access Management (IAM) | Update Account Alias

- Identity and Access Management (IAM) | Update Entity or Policy (Read-Write Permissions)
- Identity and Access Management (IAM) | Update Entity or Policy (Review Required)
- Identity and Access Management (IAM) | Update MaxSessionDuration
- Identity and Access Management (IAM) | Update SAML Identity Provider
- KMS Alias | Delete
- KMS Key | Delete (Review Required)
- KMS Key | Enable Rotation
- KMS Key | Share (Review Required)
- KMS Key | Update (Review Required)
- Load Balancer (ELB) Stack | Replace Listener Certificate
- Load Balancer (ELB) Stack | Update
- Network Load Balancer | Add Listener Certificate
- Network Load Balancer | Remove Listener Certificate
- Network Load Balancer | Update
- RDS Database Stack | Reboot
- RDS Database Stack | Restore To Point In Time
- RDS Database Stack | Rotate DB Certificate
- RDS Database Stack | Start Aurora Cluster
- RDS Database Stack | Start DB Instance
- RDS Database Stack | Stop Aurora Cluster
- RDS Database Stack | Stop DB Instance
- RDS Database Stack | Update
- RDS Database Stack | Update (For Aurora)
- RDS Database Stack | Update Deletion Protection
- RDS Database Stack | Update Enhanced Monitoring
- RDS Database Stack | Update Instance Type
- RDS Database Stack | Update Maintenance Window (Review Required)
- RDS Database Stack | Update Master User Password
- RDS Database Stack | Update MultiAZ Setting

- RDS Database Stack | Update Performance Insights (Review Required)
- RDS Database Stack | Update Storage
- RDS Snapshot | Delete
- RDS Snapshot | Share
- Redshift | Pause Cluster
- Redshift | Resume Cluster
- Route 53 Resolver | Associate VPC With Resolver Rule
- Route 53 Resolver | Disassociate Resolver Rules from VPC
- S3 Storage | Add Event Notification
- S3 Storage | Add Replication Rule
- S3 Storage | Delete Policy (Review Required)
- S3 Storage | Manage Lifecycle Configuration
- S3 Storage | Receive Replication Replica
- S3 Storage | Update
- S3 Storage | Update Encryption
- S3 Storage | Update Policy (Review Required)
- S3 Storage | Update Versioning
- Security Group | Associate
- Security Group | Authorize Egress Rule
- Security Group | Authorize Ingress Rule
- Security Group | Delete
- Security Group | Delete (Review Required)
- Security Group | Disassociate
- Security Group | Revoke Egress Rule
- Security Group | Revoke Ingress Rule
- Security Group | Update (Review Required)
- Stack | Delete
- Stack Patching Configuration | Update
- Tag | Bulk Update
- Tag | Bulk Update (Review Required)

- Tag | Delete
- Tag | Delete (Review Required)
- Tag | Update
- Tag | Update (Review Required)
- Target Group | Attach Instances
- Target Group | Detach Instances
- Target Group | Update (For ALB)
- Target Group | Update (For NLB)

ACM | Delete Certificate

Delete an AWS Certificate Manager (ACM) certificate that is currently not in use and not managed by AMS.

Full classification: Management | Advanced stack components | ACM | Delete certificate

Change Type Details

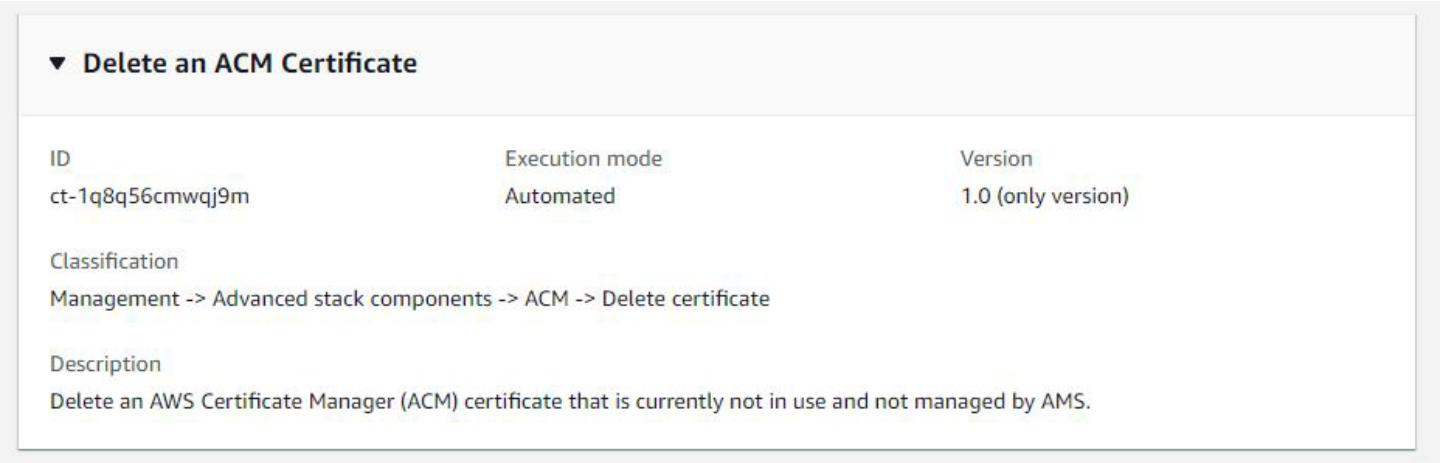
Change type ID	ct-1q8q56cmwqj9m
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete ACM certificate

Deleting an ACM with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an ACM certificate with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1q8q56cmwqj9m" --change-type-version "1.0"
 --title "Delete an ACM Certificate" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-DeleteACMCertificate\",\"Region\": \"us-east-1\",\"Parameters
\": {\"CertificateARN\": [\"arn:aws:acm:us-east-1:123456789012:certificate/c96c73cd-
d082-4fa9-bbf2-09d8600d84ad\"]}}"
```

TEMPLATE CREATE:

1. Save the execution parameters to a JSON file named DeleteCertificateParameters.json.

```
{
    "DocumentName": "AWSManagedServices-DeleteACMCertificate",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateARN": [
            "arn:aws:acm:us-east-1:123456789012:certificate/c96c73cd-d082-4fa9-
bbf2-09d8600d84ad"
        ]
    }
}
```

2. Output the RFC template to a file in your current folder; this example names it DeleteCertificateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteCertificateRfc.json
```

3. Modify and save the DeleteCertificateRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":         "ct-1q8q56cmwqj9m",
"ChangeTypeVersion":    "1.0",
"Title":                "Delete an ACM Certificate"
}
```

4. Create the RFC, specifying the CreateAcmRfc file and the CreateAcmParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteCertificateRfc.json  --
execution-parameters file://DeleteCertificateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about ACM certificates, see What Is AWS Certificate Manager? and ACM Certificate Characteristics.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1q8q56cmwqj9m.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteACMCertificate",
  "Region" : "us-east-1",
  "Parameters": {
    "CertificateARN": ["arn:aws:acm:us-
east-1:111111111111:certificate/1111aaaa-11aa-11aa-11aa-111111aaaaaa"]
  }
}
```

AMI | Deregister

Deregister one or multiple Amazon Machine Images (AMI)s and optionally delete all associated snapshots. Once deregistered the AMI or AMIs can't be used for launching new instances.

Full classification: Management | Advanced stack components | AMI | Deregister

Change Type Details

Change type ID	ct-26vhhlj9jmlpf

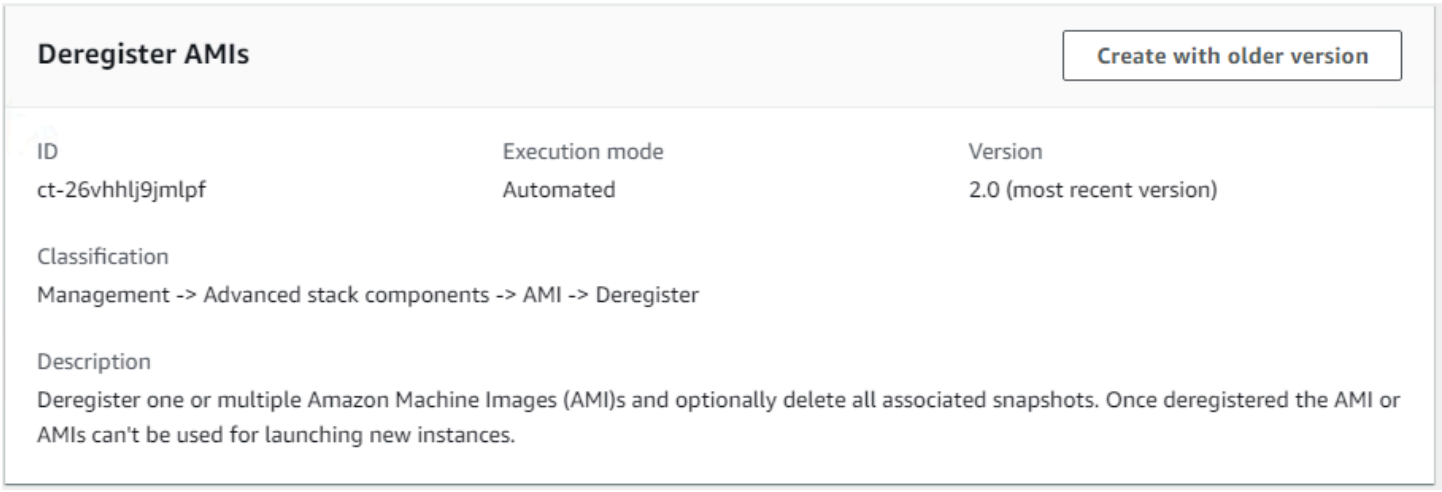
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete or deregister multiple AMIs

Deregistering AMIs with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deregistering AMIs with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \

 --change-type-id "ct-26vhhlj9jmlpf" \
 --change-type-version "2.0" --title "Deregister multiple AMIs" \
   --execution-parameters "{\"DocumentName\":\"AWSManagedServices-
BulkDeleteOrDeregisterAMI\",\"Region\":\"us-east-1\",\"Parameters\":{\"ImageIds\":
[\"ami-0acd76831a2016e19\",\"ami-08a711de1cfa05910\"],\"DeleteSnapshots\":[false]}}"
```

> **Note**
>
> To delete instead of deregister, change the value of `DeleteSnapshots` to `[true]`.

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeregisterAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-26vhhlj9jmlpf" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeregisterAmiParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
    "DocumentName": "AWSManagedServices-BulkDeleteOrDeregisterAMI",
    "Region": "us-east-1",
    "Parameters":
      {
        "ImageIds":
          [
            "ami-0acd76831a2016e19",
            "ami-08a711de1cfa05910"
          ],
        "DeleteSnapshots":[false]
      }
}
```

3. Output the RFC template JSON file; this example names it DeregisterAmiRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeregisterAmiRfc.json
```

4. Modify and save the DeregisterAmiRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-26vhhlj9jmlpf",
  "Title": "Deregister multiple AMIs"
}
```

5. Create the RFC, specifying the DeregisterAmiRfc file and the DeregisterAmiParams file:

```
aws amscm create-rfc --cli-input-json file://DeregisterAmiRfc.json  --execution-
parameters file://DeregisterAmiParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about deregistering AMIs, see Deregister your Linux AMI.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-26vhhlj9jmlpf.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-BulkDeleteOrDeregisterAMI",
  "Region": "us-east-1",
  "Parameters" : {
    "ImageIds": [
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-BulkDeleteOrDeregisterAMI",
  "Region": "us-east-1",
  "Parameters" : {
    "ImageIds": [
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501",
      "ami-01234567891234501"
    ],
    "DeleteSnapshots": [
      false
    ]
  }
}
```

AMI | Encrypt

Use to create a custom AMI with an encrypted EBS snapshot, which protects data at rest. When the encrypted AMI is launched, the corresponding EBS volume is encrypted.

Full classification: Management | Advanced stack components | AMI | Encrypt

Change Type Details

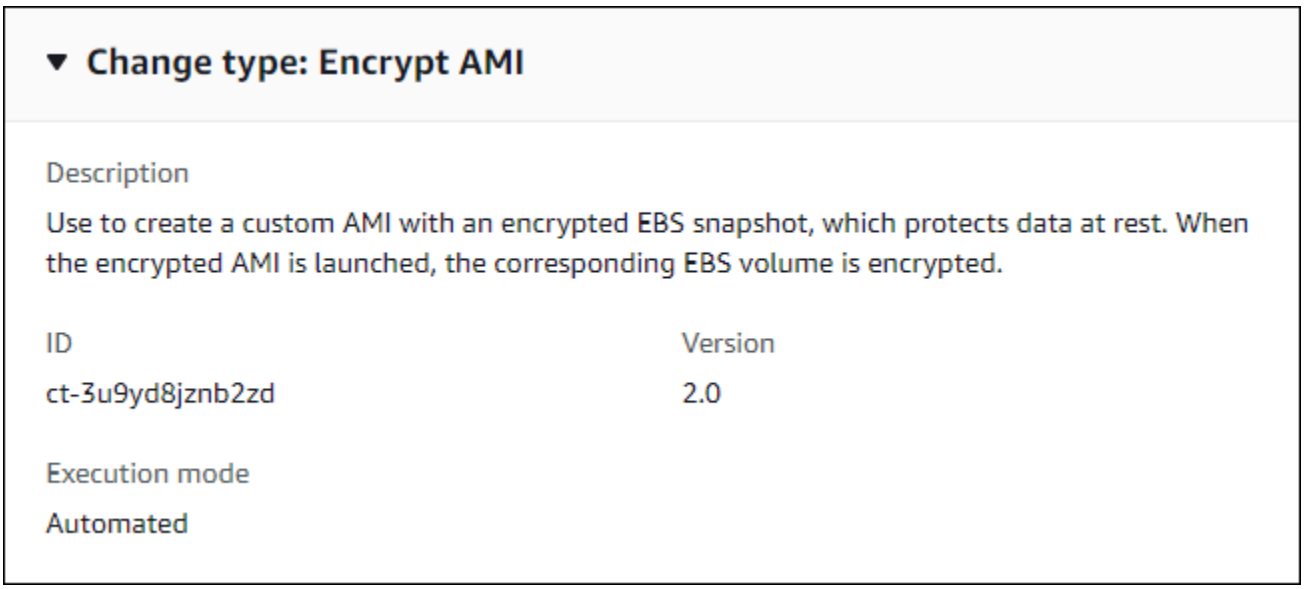
Change type ID	ct-3u9yd8jznb2zd
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Encrypt AMIs

Encrypting an AMI with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Encrypting an AMI with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc  --change-type-id
 "ct-3u9yd8jznb2zd" --change-type-version "2.0" --title "Test-AMI-Encrypt-QC" --
execution-parameters "{\"VpcId\":\"VPC_ID\", \"AmiId\":\"AMI_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it EncryptAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3u9yd8jznb2zd" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > EncryptAmiParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "VpcId":            "VPC_ID",
   "AmiId":            "AMI_ID"
   }
   ```

3. Output the RFC template JSON file; this example names it EncryptAmiRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EncryptAmiRfc.json
   ```

4. Modify and save the EncryptAmiRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-3u9yd8jznb2zd",
   "ChangeTypeVersion":    "2.0",
   "Title":                "AMI-Encrypt-RFC",
   "RequestedStartTime":   "2016-12-05T14:20:00Z",
   "RequestedEndTime":     "2016-12-05T16:20:00Z"
   }
   ```

5. Create the RFC, specifying the EncryptAmiRfc file and the EncryptAmiParams file:

```
aws amscm create-rfc --cli-input-json file://EncryptAmiRfc.json  --execution-
parameters file://EncryptAmiParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type does not support encrypting an AMI that is already encrypted.

To learn more about AWS AMI encryption, see AMIs with Encrypted Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3u9yd8jznb2zd.

Example: Required Parameters

```
{
  "AmiId" : "ami-01234567890abcdef",
  "VpcId" : "vpc-01234567890abcdef"
}
```

Example: All Parameters

```
{
  "AmiId" : "ami-12345678",
  "VpcId" : "vpc-12345678",
  "KmsKeyId": "a3ccc020-abcd-1234-8d69-2f060c3c1234"
}
```

AMI | Share

Use to share an AMI with another AMS account.

Full classification: Management | Advanced stack components | AMI | Share

Change Type Details

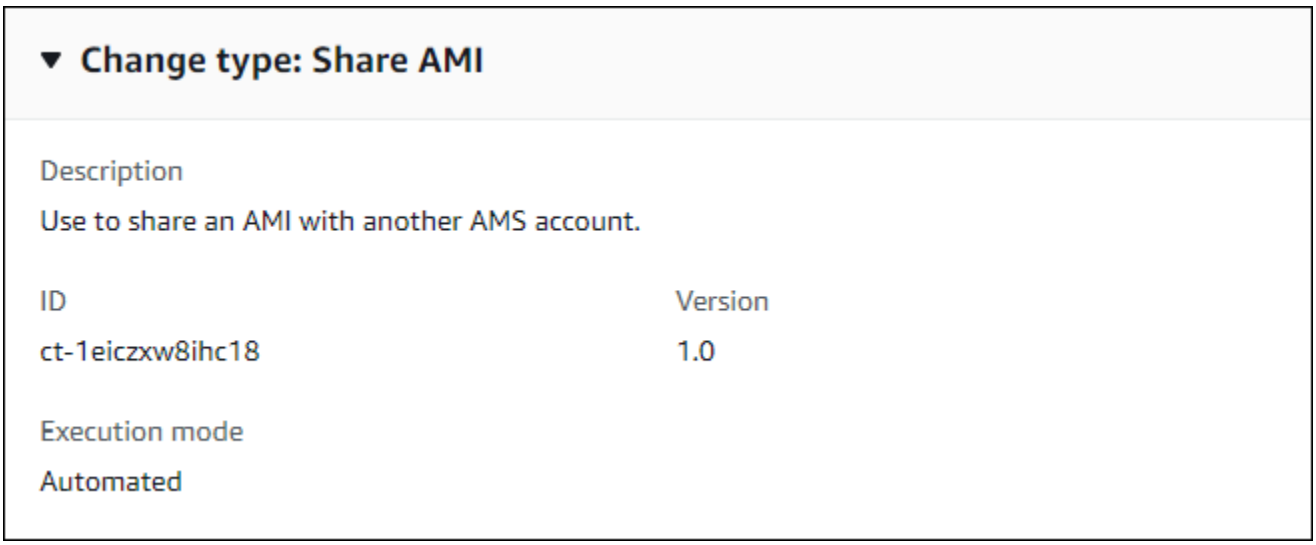
Change type ID	ct-1eiczxw8ihc18
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Share AMIs

Sharing an AMI with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Sharing an AMI with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml amscm create-rfc --change-type-id "ct-1eiczxw8ihc18" --change-type-
version "1.0" --title "AMI-Share-QC" --execution-parameters "{\"Description\":\"Share
 custom AMI\", \"AmiId\":\"AMI-ID\", \"TargetAwsAccountId\":, \"AWS-ACCOUNT-ID\":}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ShareAmiParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1eiczxw8ihc18" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > ShareAmiParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "TargetAwsAccountId":   "AMS_ACCOUNT_ID",
   "AmiId":                "AMI_ID"
   }
   ```

3. Output the RFC template JSON file; this example names it EncryptAmiRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > EncryptAmiRfc.json
```

4. Modify and save the EncryptAmiRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-1eiczxw8ihc18",
"ChangeTypeVersion":     "1.0",
"Title":                 "AMI-Share-RFC"
}
```

5. Create the RFC, specifying the ShareAmiRfc file and the ShareAmiParams file:

```
aws amscm create-rfc --cli-input-json file://ShareAmiRfc.json  --execution-
parameters file://ShareAmiParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> An AMS AMI can't be shared to a non-AMS account. This change type copies an AMI from the source AWS Region to the same Region in the destination account. You must be onboarded to AMS in the destination AWS Region of the specified TargetAwsAccountId, or the shared AMI is unusable in the target account.
> Additionally, encrypted AMIs can't be shared between accounts without the involvement of the AD administrator and an AMS Operations Engineer. If you want to do this, file a Management | Other | Other | Create (ct-1e1xtak34nx76) with AMS with the AMI ID, account information, and full details.

Important

When sharing a custom AMI that you created from an instance in your AMS account, ensure that the AMI is usable in the new account. In particular, the instance used to create the AMI must have been separated from its domain. For more information, see Create an AMI.

To learn more about sharing AMIs, see Sharing an AMI with Specific AWS Accounts.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1eiczxw8ihc18.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "AmiId": "ami-12345678",
  "TargetAwsAccountId": "123456789012"
}
```

Application Load Balancer | Add Listener Certificate

Add a certificate to the specified Application Load Balancer (ALB) listener. Use the RemediateStackDrift parameter for the automation to try to remediate drift, if it is introduced.

Full classification: Management | Advanced stack components | Application Load Balancer | Add listener certificate

Change Type Details

Change type ID	ct-3g6fq83nxg1a7
Current version	1.0
Expected execution duration	60 minutes

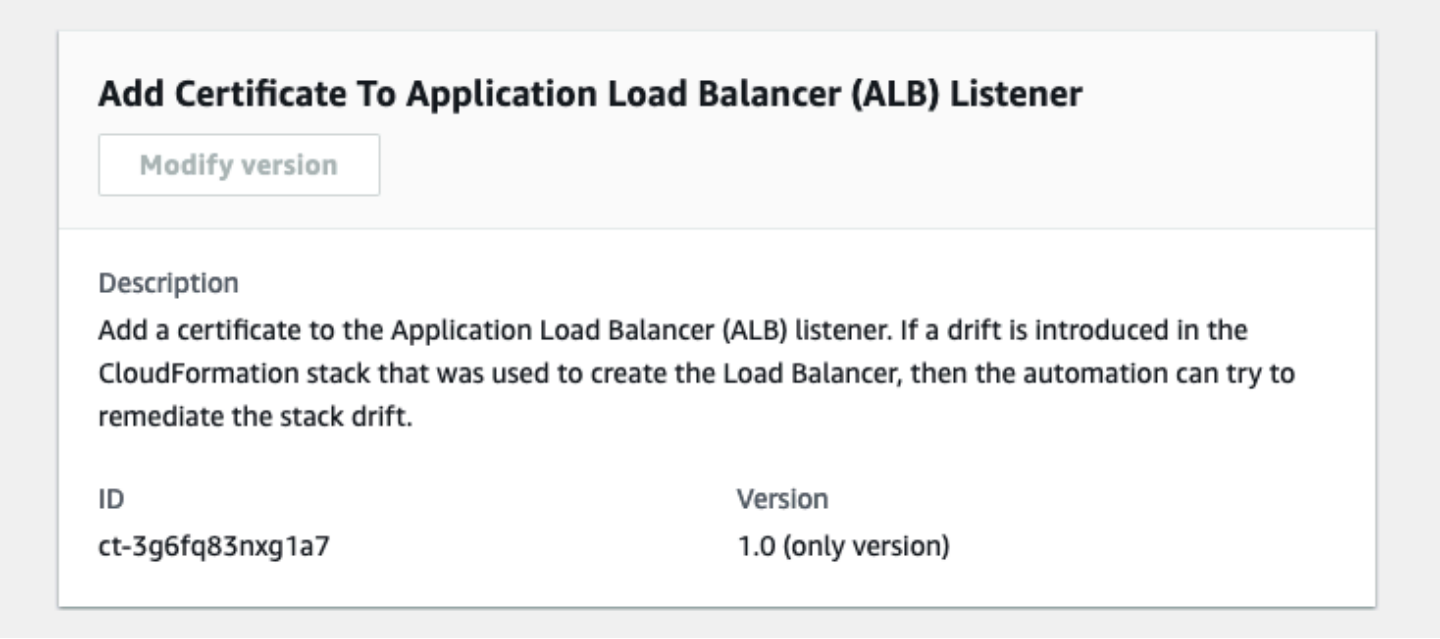
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add ALB listener certificate

Adding a listener certificate to an ALB with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a listener certificate to an ALB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-3g6fq83nxg1a7" --change-type-version
 "1.0" --title "Add listener certificate ALB" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-AddCertificateToElbv2Listener\",\"Region\": \"us-
east-1\",\"Parameters\":{\"ListenerArn\":[\"arn:aws:elasticloadbalancing:us-
east-1:123456789012:listener/app/testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e\"],
\"CertificateArn\":[\"arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba\"],\"IsDefault\":[\"False\"],\"RemediateStackDrift
\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3g6fq83nxg1a7"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AddAlbListenerCertParams.json
   ```

2. Modify and save the AddAlbListenerCertParams file. For example:

   ```
   {
       "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
       "Region": "us-east-1",
       "Parameters": {
   ```

```
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e"
        ],
        "CertificateArn": [
            "arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba"
        ],
        "IsDefault": [
            "False"
        ],
        "RemediateStackDrift": [
            "True"
        ]
    }
}
```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > AddAlbListenerCertRfc.json
```

4. Modify and save the AddAlbListenerCertRfc.json file. For example:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3g6fq83nxg1a7",
"Title":                "ALB-Add-Listener-Cert-RFC"
}
```

5. Create the RFC, specifying the AddAlbListenerCertRfc file and the AddAlbListenerCertParams file:

```
aws amscm create-rfc --cli-input-json file://AddAlbListenerCertRfc.json --
execution-parameters file://AddAlbListenerCertParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Application Load Balancers, see What Is an Application Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3g6fq83nxg1a7.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "IsDefault": [
            "True"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ],
```

```
        "RemediateStackDrift": [
            "False"
        ]
    }
}
```

Application Load Balancer | Remove Listener Certificate

Remove a certificate from the specified Application Load Balancer (ALB) listener. Use the RemediateStackDrift parameter for the automation to try to remediate drift, if it is introduced.

Full classification: Management | Advanced stack components | Application Load Balancer | Remove listener certificate

Change Type Details

Change type ID	ct-0tpbr6lfa3zng
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove ALB listener certificate

Removing a listener certificate from an ALB with the console

The following shows this change type in the AMS console.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing a listener certificate from an ALB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-0tpbr6lfa3zng" --change-type-version "1.0"
 --title "Remove listener certificate ALB" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-RemoveCertificateToElbv2Listener\",\"Region\": \"us-
east-1\",\"Parameters\":{\"ListenerArn\":[\"arn:aws:elasticloadbalancing:us-
east-1:123456789012:listener/app/testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e\"],
\"CertificateArn\":[\"arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba\"],\"IsDefault\":[\"False\"],\"RemediateStackDrift
\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0tpbr6lfa3zng"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RemoveAlbListenerCertParams.json
   ```

2. Modify and save the RemoveAlbListenerCertParams file. For example:

```
{
    "DocumentName": "AWSManagedServices-RemoveCertificateToElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e"
        ],
        "CertificateArn": [
            "arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba"
        ],
        "IsDefault": [
            "False"
        ],
        "RemediateStackDrift": [
            "True"
        ]
    }
}
```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > RemoveAlbListenerCertRfc.json
```

4. Modify and save the RemoveAlbListenerCertRfc.json file. For example:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0tpbr6lfa3zng",
"Title":                "ALB-Remove-Listener-Cert-RFC"
}
```

5. Create the RFC, specifying the RemoveAlbListenerCertRfc file and the RemoveAlbListenerCertParams file:

```
aws amscm create-rfc --cli-input-json file://RemoveAlbListenerCertRfc.json --
execution-parameters file://RemoveAlbListenerCertParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Application Load Balancers, see What Is an Application Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0tpbr6lfa3zng.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-RemoveCertificateFromElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-RemoveCertificateFromElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/app/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
```

```
        ],
        "RemediateStackDrift": [
            "False"
        ]
    }
}
```

Application Load Balancer | Update

Update the properties of an existing AWS Application Load Balancer (ALB) that was created by version 3.0 CT: ct-111r1yayblnw4.

Full classification: Management | Advanced stack components | Application Load Balancer | Update

Change Type Details

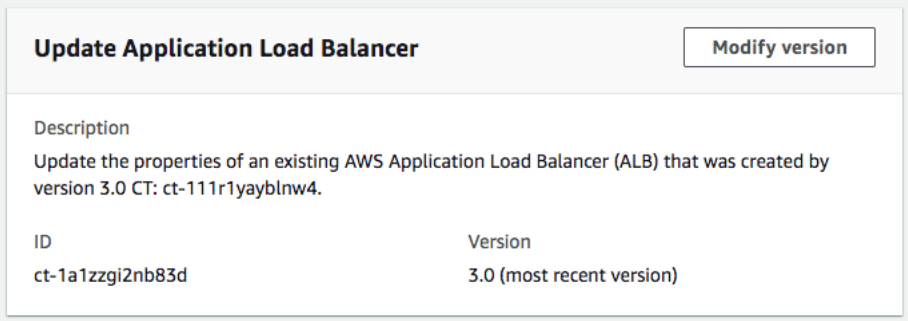
Change type ID	ct-1a1zzgi2nb83d
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update application load balancer (ALB)

Updating an ALB with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an ALB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title Test-Update-ALB --change-type-id ct-1a1zzgi2nb83d
 --change-type-version 3.0 --execution-parameters '{"Description":"Updating Test
 ALB","VpcId":"VPC_ID","StackTemplateId":"stm-sd7uv500000000000","Name":"Test-
Application-LoadBalancer","TimeoutInMinutes":360,"Parameters":
{"TargetGroupHealthCheckPath": "/myAppHealth"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

```
aws amscm get-change-type-version --change-type-id "ct-111r1yayblnw4" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateAlbParams.json
```

2. Modify and save the UpdateAlbParams file. For example:

```
{
"Description":      "ALB-Update",
"VpcId":            "VPC_ID",
"Name":             "My-ALB",
"StackTemplateId":  "stm-sd7uv500000000000",
"TimeoutInMinutes" : 360,
"Parameters": {
    "LoadBalancerSecurityGroups": [
      "sg-1234567890abcdef0"
    ],
    "LoadBalancerSubnetIds": [
      "subnet-1234567890abcdef0",
      "subnet-1234567890abcdef1"
    ],
    "LoadBalancerDeletionProtection": "false",
    "LoadBalancerIdleTimeout": "60",
    "Listener1Port": "443",
    "Listener1Protocol": "HTTPS",
    "Listener1SSLCertificateArn": "arn:aws:acm:ap-
southeast-2:012345678912:certificate/e23c3545-e92d-4542-83b8-63483505b5a5",
    "Listener1SSLPolicy": "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
    "Listener2Port": "8080",
    "Listener2Protocol": "HTTP",
    "TargetGroupHealthCheckInterval": "10",
    "TargetGroupHealthCheckPath": "/thing/index.html",
    "TargetGroupHealthCheckPort": "8080",
    "TargetGroupHealthCheckProtocol": "HTTP",
    "TargetGroupHealthCheckTimeout": "10",
    "TargetGroupHealthyThreshold": "2",
    "TargetGroupUnhealthyThreshold": "10",
    "TargetGroupValidHTTPCode": "200",
    "TargetGroupDeregistrationDelayTimeout": "300",
    "TargetGroupSlowStartDuration": "30",
```

```
    "TargetGroupCookieExpirationPeriod": "20"
  }
}
```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateAlbRfc.json
```

4. Modify and save the UpdateAlbRfc.json file. For example:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-111r1yayblnw4",
"Title":                "ALB-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateAlbRfc file and the UpdateAlbParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateAlbRfc.json --execution-
parameters file://UpdateAlbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is version 3.0, and can be used with the version 3.0 of the Create ALB change type (ct-111r1yayblnw4).

To learn more about AWS Application Load Balancers, see What Is an Application Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1a1zzgi2nb83d.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
    "LoadBalancerSecurityGroups": ["sg-12345678"],
    "LoadBalancerSubnetIds": ["subnet-12345678", "subnet-12345688"],
    "LoadBalancerDeletionProtection": "false",
    "LoadBalancerIdleTimeout": "60",
    "Listener1Port": "443",
    "Listener1Protocol": "HTTPS",
    "Listener1SSLCertificateArn": "arn:aws:acm:ap-southeast-2:012345678912:certificate/
e23c3545-e92d-4542-83b8-63483505b5a5",
    "Listener1SSLPolicy": "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
    "Listener2Port": "8080",
    "Listener2Protocol": "HTTP",
    "TargetGroupHealthCheckInterval": "10",
    "TargetGroupHealthCheckPath": "/thing/index.html",
    "TargetGroupHealthCheckPort": "8080",
    "TargetGroupHealthCheckProtocol": "HTTP",
    "TargetGroupHealthCheckTimeout": "10",
    "TargetGroupHealthyThreshold": "2",
    "TargetGroupUnhealthyThreshold": "10",
    "TargetGroupValidHTTPCode": "200",
    "TargetGroupDeregistrationDelayTimeout": "300",
    "TargetGroupSlowStartDuration": "30",
    "TargetGroupCookieExpirationPeriod": "20"
  }
}
```

Auto Scaling Group | Update

Update an Auto Scaling Group and associated launch configuration created with CT ct-2tylseo8rxfsc, version 2.0.

Full classification: Management | Advanced stack components | Auto scaling group | Update

Change Type Details

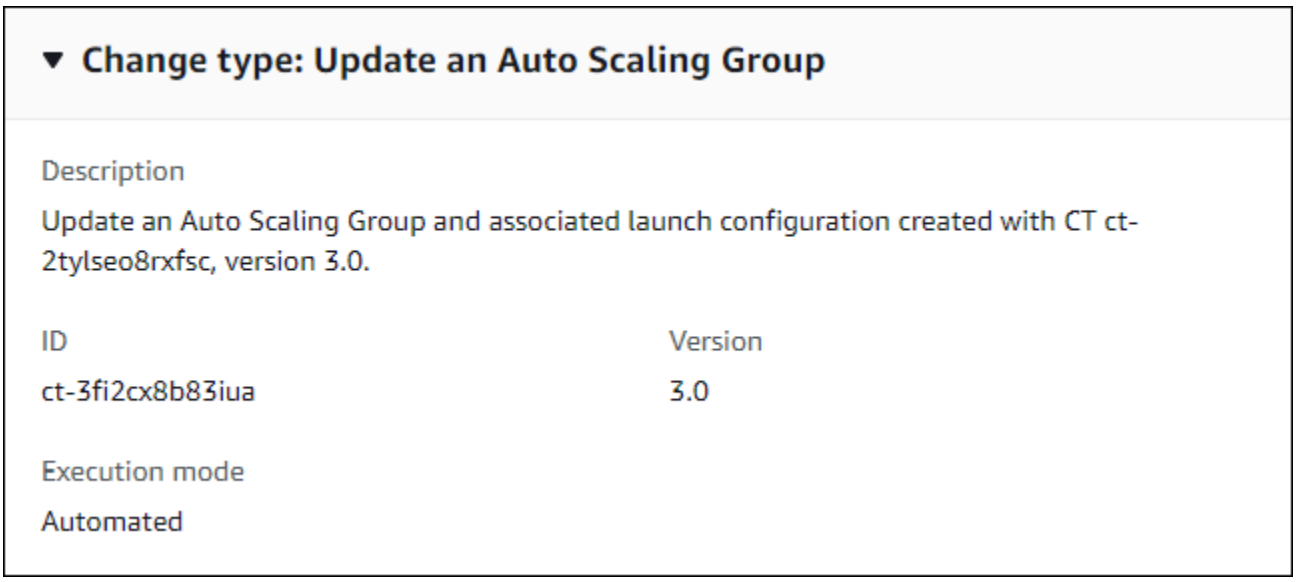
Change type ID	ct-3fi2cx8b83iua
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update Auto Scaling groups

Updating an Auto Scaling group with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an Auto Scaling group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  --profile saml --region us-east-1 create-rfc --change-type-id
 "ct-3fi2cx8b83iua" --change-type-version "2.0" --title "Test-Update ASG" --description
 "Test Update"  --execution-parameters "{\"VpcId\":\"VPC_ID\",\"StackId\":\"STACK_ID\",
\"Parameters\":{\"ASGAmiId\":\"AMI_ID\",\"ASGInstanceType\":\"m3.medium\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it UpdateAsgParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3fi2cx8b83iua" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateAsgParams.json
   ```

   > **Note**
   >
   > Scripts are newline-delimited (separate with literal: "\n"), also, scripts entered as UserData are executed as the "root" user and do not need to use the "sudo" command. The RFC waits up to six hours for all of the UserData script commands to execute before returning a final status of success or failure.

2. Modify and save the file. For example, you can replace the contents with something like this:

   ```
   {
     "VpcId": "VPC_ID",
     "StackId": "STACK_ID",
     "Parameters": {
       "ASGAmiId": "AMI_ID",
       "ASGInstanceType": "m3.medium"
     }
   }
   ```

3. Output the JSON template for UpdateRfc to a file in your current folder; example names it UpdateAsgRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateAsgRfc.json
   ```

4. Modify and save the JSON file as follows. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-3fi2cx8b83iua",
"Title":                "ASG-Update-Stack-RFC"
}
```

5. Create the RFC, specifying the UpdateAsgRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://UpdateAsgRfc.json --execution-
parameters file://UpdateAsgParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a version 2.0 change type and can be used to update Auto Scaling groups A(SG) created with the corresponding version 2.0 create change type, ct-2tylseo8rxfsc.

To learn more, see Amazon Auto Scaling.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3fi2cx8b83iua.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
```

```
  "VpcId": "vpc-12345678",
  "StackId": "stack-12345678901234567",
  "Parameters": {
    "ASGAmiId": "ami-a0b1c2d3",
    "ASGCooldown": 300,
    "ASGDesiredCapacity": 1,
    "ASGEBSOptimized": "false",
    "ASGIAMInstanceProfile": "customer-mc-ec2-instance-profile",
    "ASGInstanceDetailedMonitoring": "false",
    "ASGInstanceRootVolumeIops": 0,
    "ASGInstanceRootVolumeSize": 8,
    "ASGInstanceRootVolumeType": "standard",
    "ASGInstanceType": "m3.medium",
    "ASGLoadBalancerNames": ["elb1"],
    "ASGMaxInstances": 1,
    "ASGMinInstances": 1,
    "ASGHealthCheckGracePeriod": 600,
    "ASGHealthCheckType":"EC2",
    "ASGScaleDownMetricName": "CPUUtilization",
    "ASGScaleDownPolicyCooldown": 300,
    "ASGScaleDownPolicyEvaluationPeriods": 4,
    "ASGScaleDownPolicyPeriod": 60,
    "ASGScaleDownPolicyScalingAdjustment": -1,
    "ASGScaleDownPolicyStatistic": "Average",
    "ASGScaleDownPolicyThreshold": 35,
    "ASGScaleUpMetricName": "CPUUtilization",
    "ASGScaleUpPolicyCooldown": 60,
    "ASGScaleUpPolicyEvaluationPeriods": 2,
    "ASGScaleUpPolicyPeriod": 60,
    "ASGScaleUpPolicyScalingAdjustment": 2,
    "ASGScaleUpPolicyStatistic": "Average",
    "ASGScaleUpPolicyThreshold": 75,
    "ASGSubnetIds": ["subnet-a0b1c2d3", "subnet-e4f5g6h7"],
    "ASGUserData": "#!/bin/bash\npwd\nls -ltrh\necho \"Hello, World\""
  }
}
```

Bastions | Add CIDR Ingress (Review Required)

Add RDP or SSH bastion ingress Classless Inter-Domain Routing (CIDR) allow lists.

Full classification: Management | Advanced stack components | Bastions | Add CIDR ingress (review required)

Change Type Details

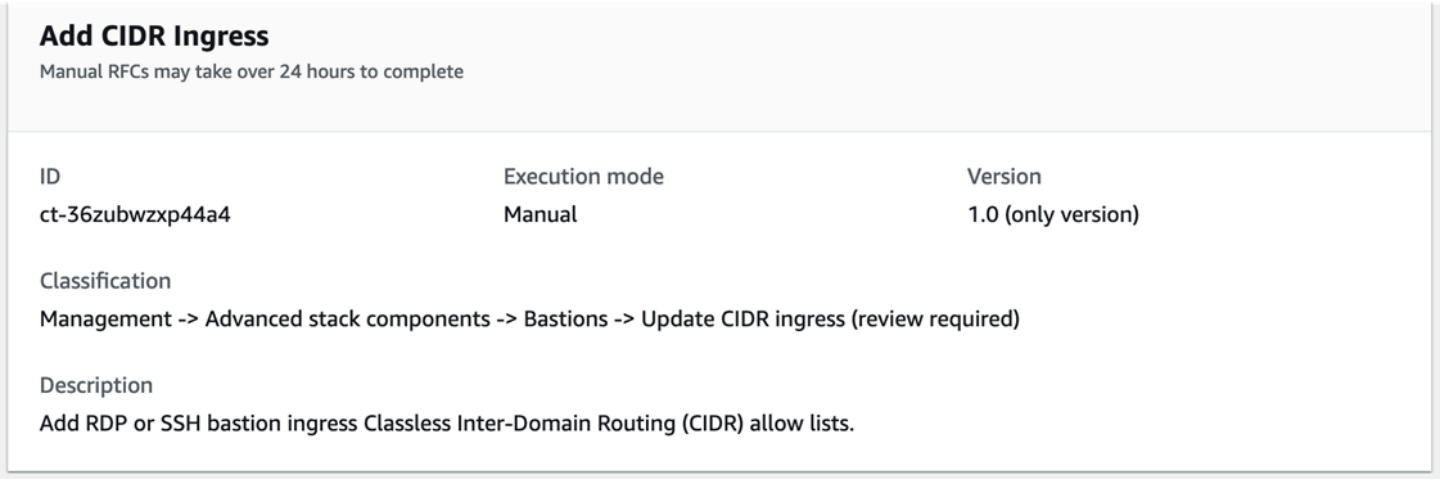
Change type ID	ct-36zubwzxp44a4
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Add CIDR ingress (review required) ct-36zubwzxp44a4

Adding bastion CIDR ingress with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding bastion CIDR ingress with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-36zubwzxp44a4" --change-type-version "1.0"
 --title "Add CIDR ingress" --execution-parameters "{\"BastionType\": \"RDP Bastion\",
 \"[\"10.0.0.1/24\",\"10.20.0.4/25\",\"10.0.0.6/25\"]\": \"10\", \"ASGMinCount\":
 \"10\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it AddBastionCidrIngressParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-36zubwzxp44a4"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AddBastionCidrIngressParams.json
   ```

2. Modify and save the AddBastionCidrIngressParams file.

   ```
   {
      "BastionType": "RDP Bastion",
      "IngressCIDRAddresses": ["10.113.44.1/22", "10.113.56.1/22"],
      "Priority": "Medium"
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it AddBastionCidrIngressRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AddBastionCidrIngressRfc.json
```

4. Modify and save the AddBastionCidrIngressRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-36zubwzxp44a4",
  "Title": "Add Bastion CIDR Ingress"
}
```

5. Create the RFC, specifying the AddBastionCidrIngressRfc file and the AddBastionCidrIngressParams file:

```
aws amscm create-rfc --cli-input-json file://AddBastionCidrIngressRfc.json --
execution-parameters file://AddBastionCidrIngressParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- To learn more, see Authorize inbound traffic for your Linux instances.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-36zubwzxp44a4.

Example: Required Parameters

```
{
  "BastionType": "RDP Bastion",
  "IngressCIDRAddresses": ["10.113.44.1/22"]
}
```

Example: All Parameters

```
{
  "BastionType": "RDP Bastion",
  "IngressCIDRAddresses": ["10.113.44.1/22", "10.113.56.1/22"],
  "Priority": "Medium"
}
```

Bastions | Update Instance or Session Counts (Review Required)

Update the number of RDP and SSH Bastion instances. Optionally update the session count of RDP Bastions.

Full classification: Management | Advanced stack components | Bastions | Update instance or session counts (review required)

Change Type Details

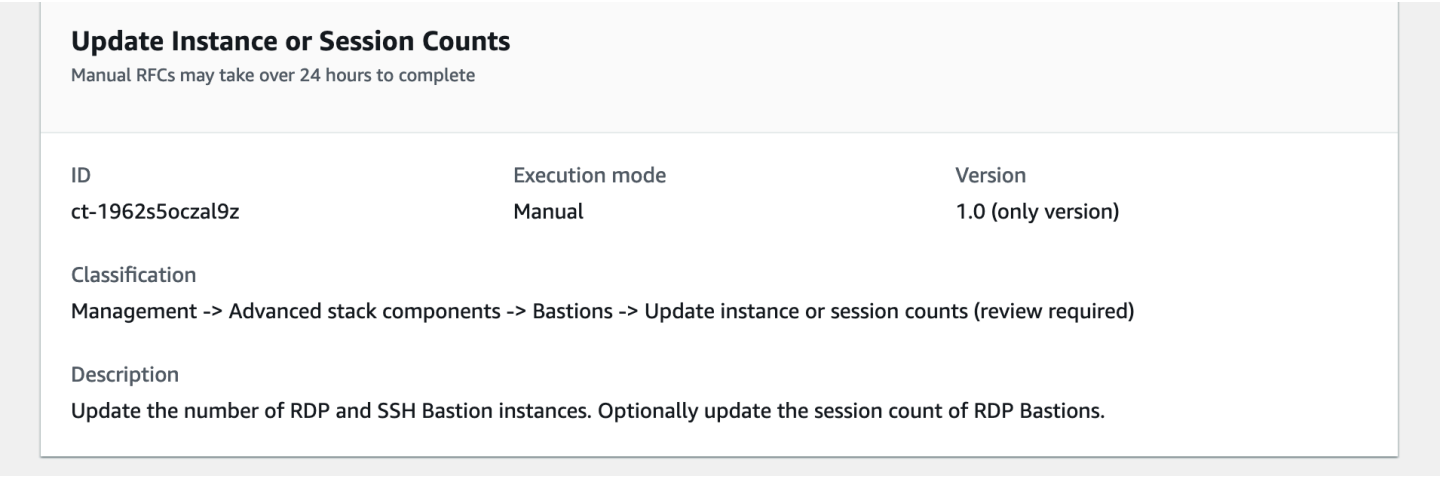
Change type ID	ct-1962s5oczal9z
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update bastion instance or session counts (review required) ct-1962s5oczal9z

Updating bastion instance or session counts with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating bastion instance or session counts with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

SSH Bastion Update Example

```
aws amscm create-rfc --change-type-id "ct-1962s5oczal9z" --change-type-version "1.0"
 --title "Update instance or session counts" --execution-parameters "{\"BastionType\":
 \"SSH Bastion\", \"ASGMaxCount\": \"10\", \"ASGMinCount\": \"10\", \"ASGDesiredCount
\": \"10\"}"
```

RDP Bastion Update Example

```
aws amscm create-rfc --change-type-id "ct-1962s5oczal9z" --change-type-version
 "1.0" --title "Update instance or session counts" --execution-parameters
 "{\"BastionType\": \"RDP Bastion\", \"RDPBastionDesiredMaximumSessions\": \"50\",
 \"RDPBastionDesiredMinimumSessions\": \"20\"}"
```

RDP Bastion Update Example with ASG

```
aws amscm create-rfc --change-type-id "ct-1962s5oczal9z" --change-type-version "1.0"
 --title "Update instance or session counts" --execution-parameters "{\"BastionType\":
 \"RDP Bastion\", \"ASGMaxCount\": \"10\", \"ASGMinCount\": \"10\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateBastionInstSesCountsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1962s5oczal9z"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateBastionInstSesCountsParams.json
   ```

2. Modify and save the UpdateBastionInstSesCountsParams file.

   ```
   {
     "BastionType": "RDP Bastion",
     "RDPBastionDesiredMaximumSessions": 10,
   ```

```
   "RDPBastionDesiredMinimumSessions": 2,
   "ASGMaxCount": 4,
   "ASGMinCount": 0,
   "ASGDesiredCount": 2,
   "Priority": "Medium"
 }
```

3. Output the RFC template to a file in your current folder; this example names it UpdateBastionInstSesCountsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateBastionInstSesCountsRfc.json
```

4. Modify and save the UpdateBastionInstSesCountsRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1962s5oczal9z",
  "Title": "Update Bastion Instance or Session Count"
}
```

5. Create the RFC, specifying the UpdateBastionInstSesCountsRfc file and the UpdateBastionInstSesCountsParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateBastionInstSesCountsRfc.json --
execution-parameters file://UpdateBastionInstSesCountsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- To learn more about AWS bastions, see Linux Bastion Hosts on AWS.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1962s5oczal9z.

Example: Required Parameters

```
{
  "BastionType": "RDP Bastion"
}
```

Example: All Parameters

```
{
  "BastionType": "RDP Bastion",
  "RDPBastionDesiredMaximumSessions": 10,
  "RDPBastionDesiredMinimumSessions": 2,
  "ASGMaxCount": 4,
  "ASGMinCount": 0,
  "ASGDesiredCount": 2,
  "Priority": "Medium"
}
```

Bastions | Update Instance Size (Review Required)

Update the instance size for an RDP or SSH customer bastion in an AMS account.

Full classification: Management | Advanced stack components | Bastions | Update instance size (review required)

Change Type Details

Change type ID	ct-2x14cv67uym46
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter

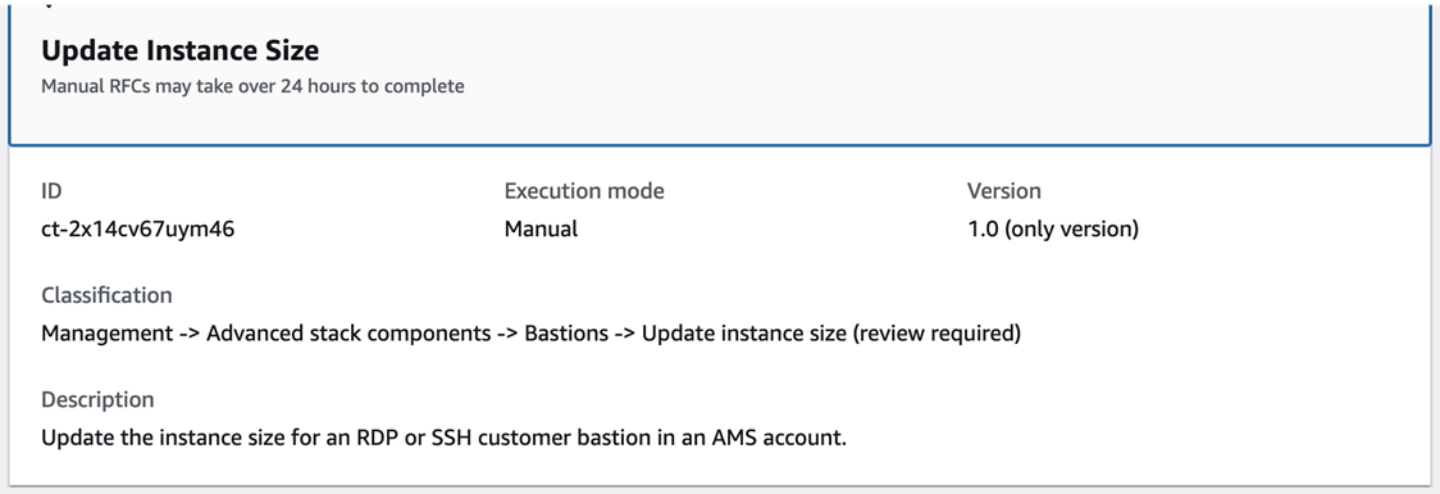
Execution mode	Manual

Additional Information

Update bastion instance size (review required) ct-2x14cv67uym46

Updating bastion instance size with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating bastion instance size with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2x14cv67uym46" --change-type-version "1.0" --
title "Update instance size" --execution-parameters "{\"InstanceType\": \"t3.medium\",
 \"BastionType\": \"RDP Bastion\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateBastionInstSizeParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2x14cv67uym46"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateBastionInstSizeParams.json
   ```

2. Modify and save the UpdateBastionInstSizeParams file.

   ```
   {
      "BastionType": "RDP Bastion",
      "InstanceType": "t3.medium",
      "Priority": "Medium"
    }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdateBastionInstSizeRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateBastionInstSizeRfc.json
   ```

4. Modify and save the UpdateBastionInstSizeRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
   ```

```
  "ChangeTypeId": "ct-2x14cv67uym46",
  "Title": "Update Bastion Instance Size"
}
```

5. Create the RFC, specifying the UpdateBastionInstSizeRfc file and the UpdateBastionInstSizeParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateBastionInstSizeRfc.json --
execution-parameters file://UpdateBastionInstSizeParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- To learn more about AWS bastions, see Linux Bastion Hosts on AWS.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2x14cv67uym46.

Example: Required Parameters

```
{
  "BastionType": "RDP Bastion",
  "InstanceType": "t3.medium"
}
```

Example: All Parameters

```
{
  "BastionType": "RDP Bastion",
  "InstanceType": "t3.medium",
```

```
  "Priority": "Medium"
}
```

Database Migration Service (DMS) | Start Replication Task

Start a new Database Migration Service (DMS) replication task, or a task in a stopped or failed state.

Full classification: Management | Advanced stack components | Database Migration Service (DMS) | Start replication task

Change Type Details

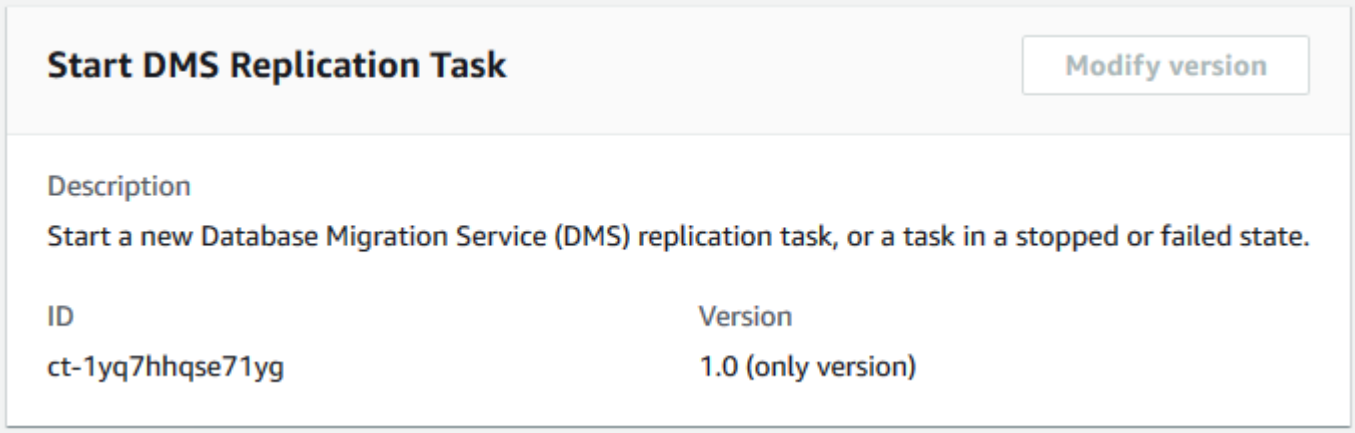
Change type ID	ct-1yq7hhqse71yg
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start AWS DMS replication task

Starting a AWS DMS replication task with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting a AWS DMS replication task with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1yq7hhqse71yg" --change-type-version
 "1.0" --title "Start DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StartDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"],\"StartReplicationTaskType\":[\"start-
replication\"],\"CdcStartPosition\":[\"\"],\"CdcStopPosition\":[\"\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it StartDmsRtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1yq7hhqse71yg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartDmsRtParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-StartDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ],
         "StartReplicationTaskType": [
           "start-replication"
         ],
         "CdcStartPosition": [
           ""
         ],
         "CdcStopPosition": [
           ""
         ]
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it StartDmsRtRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StartDmsRtRfc.json
   ```

4. Modify and save the StartDmsRtRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-1yq7hhqse71yg",
  "ChangeTypeVersion": "1.0",
  "Title": "Start DMS Replication Task"
}
```

5. Create the RFC, specifying the execution parameters file and the StartDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://StartDmsRtRfc.json --execution-
parameters file://StartDmsRtParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

You can start a AWS DMS replication task, using the AMS console or the AMS API/CLI. For more information, see Working with AWS DMS Tasks.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1yq7hhqse71yg.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StartDmsTask",
  "Region": "us-east-1",
  "Parameters": {
    "ReplicationTaskArn": ["arn:aws:dms:us-
east-1:123456789000:task:3FBZMUE4QNZNMD7DMWXXOSCCM4"],
    "StartReplicationTaskType": ["start-replication"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartDmsTask",
  "Region": "us-east-1",
  "Parameters": {
    "ReplicationTaskArn": ["arn:aws:dms:us-
east-1:123456789000:task:3FBZMUE4QNZNMD7DMWXXOSCCM4"],
    "StartReplicationTaskType": ["start-replication"],
    "CdcStartPosition": ["2019-01-01T01:00:00"],
    "CdcStopPosition": ["server_time:2019-01-02T01:00:00"]
  }
}
```

Database Migration Service (DMS) | Stop Replication Task

Stop a Database Migration Service (DMS) replication task. The specified task must be in the running state.

Full classification: Management | Advanced stack components | Database Migration Service (DMS) | Stop replication task

Change Type Details

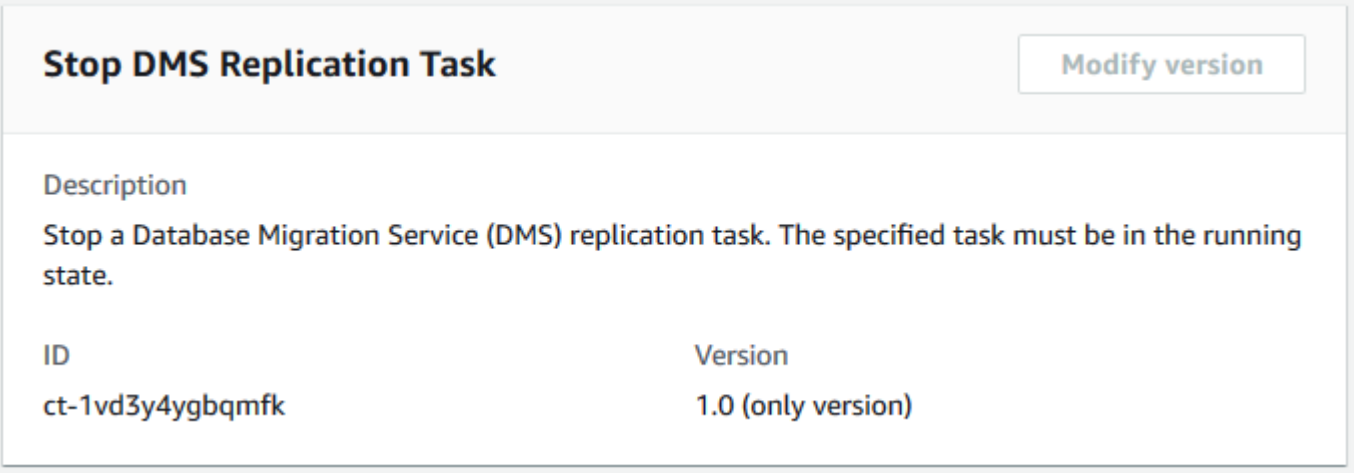
Change type ID	ct-1vd3y4ygbqmfk
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop AWS DMS replication task

Stopping a AWS DMS replication task with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping a AWS DMS replication task with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1vd3y4ygbqmfk" --change-type-version
 "1.0" --title "Stop DMS Replication Task" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StopDmsTask\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ReplicationTaskArn\":[\"TASK_ARN\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it StopDmsRtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vd3y4ygbqmfk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopDmsRtParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-StopDmsTask",
     "Region": "us-east-1",
     "Parameters": {
         "ReplicationTaskArn": [
           "TASK_ARN"
         ]
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it StopDmsRtRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopDmsRtRfc.json
   ```

4. Modify and save the StopDmsRtRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "ChangeTypeId": "ct-1vd3y4ygbqmfk",
  "ChangeTypeVersion": "1.0",
  "Title": "Stop DMS Replication Task"
}
```

5. Create the RFC, specifying the execution parameters file and the StopDmsRtRfc file:

```
aws amscm create-rfc --cli-input-json file://StopDmsRtRfc.json --execution-
parameters file://StopDmsRtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

You can stop a DMS replication task, using the AMS console or the AMS API/CLI. For more information, see Working with AWS DMS Tasks.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1vd3y4ygbqmfk.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StopDmsTask",
  "Region": "us-east-1",
  "Parameters": {
    "ReplicationTaskArn": ["arn:aws:dms:us-
east-1:123456789000:task:3FBZMUE4QNZNMD7DMWXXOSCCM4"]
  }
}
```

Directory Service | Accept Sharing

Accept a directory sharing request sent from the directory owner account. This is run in the directory consumer account.

Full classification: Management | Advanced stack components | Directory Service | Accept sharing

Change Type Details

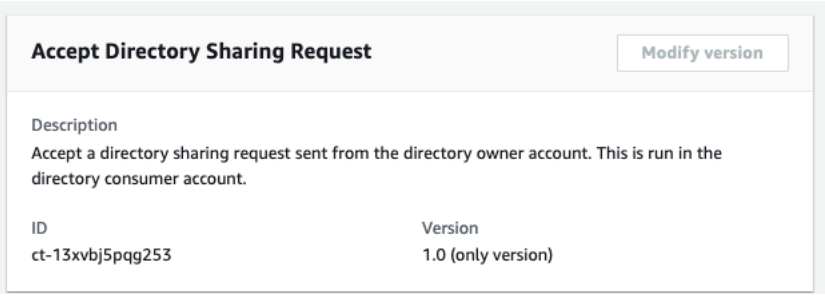
Change type ID	ct-13xvbj5pqg253
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Accept directory sharing request

Accept a directory sharing request with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Accept a directory sharing request with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \ --change-type-id "ct-13xvbj5pqg253" \ --change-type-version
 "1.0" --title "AWS Directory Service accept directory sharing" \ --execution-
parameters "{\"DocumentName\":\"AWSManagedServices-AcceptSharedDirectory\",\"Region
\":\"eu-central-1\",\"Parameters\":{\"SharedDirectoryId\":[\"d-000000000\"],
\"OwnerAccountId\":[\"000000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DirectorySharingParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-13xvbj5pqg253"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DirectorySharingParams.json
   ```

   Modify and save the DirectorySharingParams file. For example, you can replace the contents with something like this:

```
{
{
  "DocumentName": "AWSManagedServices-AcceptSharedDirectory",
  "Region": "eu-central-1",
  "Parameters": {
    "SharedDirectoryId": ["d-000000000"],
    "OwnerAccountId": ["000000000000"]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it DirectorySharingRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DirectorySharingRfc.json
```

3. Modify and save the DirectorySharingRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-13xvbj5pqg253",
"ChangeTypeVersion":     "1.0",
"Title":                 "AWS Directory Service accept directory sharing"
}
```

4. Create the RFC, specifying the DirectorySharingRfc file and the DirectorySharingParams file:

```
aws amscm create-rfc --cli-input-json file://DirectorySharingRfc.json --execution-
parameters file://DirectorySharingParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type was originally classified as Management | Advanced stack components | Directory service | Accept sharing, and has now been moved to a more user friendly classification. The change type ID, ct-13xvbj5pqg253, has not changed.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-13xvbj5pqg253.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-AcceptSharedDirectory",
  "Region": "us-east-1",
  "Parameters": {
    "SharedDirectoryId": [
      "d-12e456789f"
    ],
    "OwnerAccountId": [
      "123456789012"
    ]
  }
}
```

DNS (Private) | Update

Update an existing Route 53 DNS Hosted Zone with the supplied resource record set.

Full classification: Management | Advanced stack components | DNS (private) | Update

Change Type Details

Change type ID	ct-1d55pi44ff21u
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required

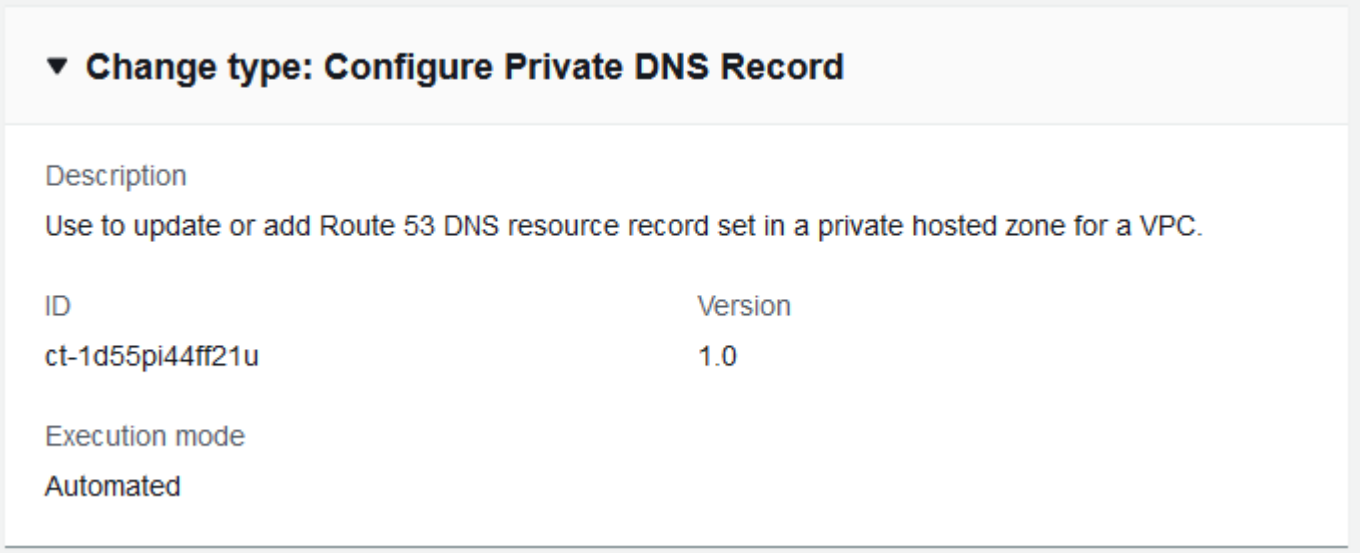
Execution mode	Automated

Additional Information

Update private DNS Route 53

Updating a private DNS Route 53 hosted zone with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a private DNS Route 53 hosted zone with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1d55pi44ff21u" \
--change-type-version "2.0" --title "Update Private DNS Record" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
CreateAddRoute53Resources\",\"Region\":\"ap-southeast-2\",\"Parameters\":{\"StackId
\":\"stack-9iwwljjcfunnrahof\",\"RecordSet\":[\"{\\\"RecordSet\\\":[{\\\"Name\\\":\\
\"test15.domain.com\\\",\\\"Type\\\":\\\"A\\\",\\\"TTL\\\":600,\\\"ResourceRecords\\\":
[\\\"10.1.1.1\\\",\\\"10.1.2.2\\\"]},{\\\"Name\\\":\\\"test16.domain.com\\\",\\\"Type\\
\":\\\"CNAME\\\",\\\"TTL\\\":600,\\\"ResourceRecords\\\":[\\\"amazon.com\\\"]}]}\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateDnsPrivateParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1d55pi44ff21u"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateDnsPrivateParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateAddRoute53Resources",
     "Region": "ap-southeast-2",
     "Parameters": {
       "HostedZoneId": "Z0188399KN0JJOZLTEXM",
       "RecordSet": [
   ```

```
      "{\"RecordSet\":[{\"Name\":\"test20.domain.com\",\"Type\":\"A\",\"TTL\":600,
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test15.domain.com\",
\"Type\":\"CNAME\",\"TTL\":600,\"ResourceRecords\":[\"amazon.com\"]}]}"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it UpdateDnsPrivateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateDnsPrivateRfc.json
```

4. Modify and save the UpdateDnsPrivateRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-1d55pi44ff21u ",
  "ChangeTypeVersion": "2.0",
  "Title": "Update Private Hosted Zone"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateDnsPrivateRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateDnsPrivateRfc.json --execution-
parameters file://UpdateDnsPrivateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- This CT fails if the specified **RecordSet** contains more than 500 resource records (RRs), or if the CloudFormation template surpasses the maximum body of 51,200 bytes.
- You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.
- To learn more, see Working with Private Hosted Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1d55pi44ff21u.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAddRoute53Resources",
  "Region" : "us-east-1",
  "Parameters": {
    "StackId": "",
    "HostedZoneId": "Z12345678901234567890",
    "RecordSet": [
      "{\"RecordSet\":[{\"Name\":\"test1.mydomain.com\",\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test3.mydomain.com
\",\"Type\":\"CNAME\",\"TTL\":\"600\",\"ResourceRecords\":[\"amazon.com\"]},
{\"Name\":\"test4.mydomain.com\",\"Type\":\"A\",\"AliasTarget\":{\"DNSName\":
\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,\"HostedZoneId
\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"weighted.mydomain.com\",\"Weight\":200,
\"SetIdentifier\":\"Example-Set-Identifier-1\",\"Type\":\"A\",\"AliasTarget\":
{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"geolocationexample.mydomain.com
\",\"SetIdentifier\":\"Example-GeoLocation-Identifier-1\",\"GeoLocation\":
{\"CountryCode\":\"US\",\"SubdivisionCode\":\"WA\"},\"Type\":\"A\",\"AliasTarget
\":{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"examplelatency.mydomain.com\",
\"SetIdentifier\":\"Example-Latency-Identifier-1\",\"Region\":\"ap-southeast-2\",
\"Type\":\"A\",\"TTL\":\"600\",\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},
{\"Name\":\"examplemultivalue.mydomain.com\",\"SetIdentifier\":\"Example-
MultiValue-Identifier-1\",\"MultiValueAnswer\":true,\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\"]}]}"
    ]
  }
}
```

DNS (Public) | Update

Update an existing Route 53 DNS Hosted Zone with the supplied resource record set.

Full classification: Management | Advanced stack components | DNS (public) | Update

Change Type Details

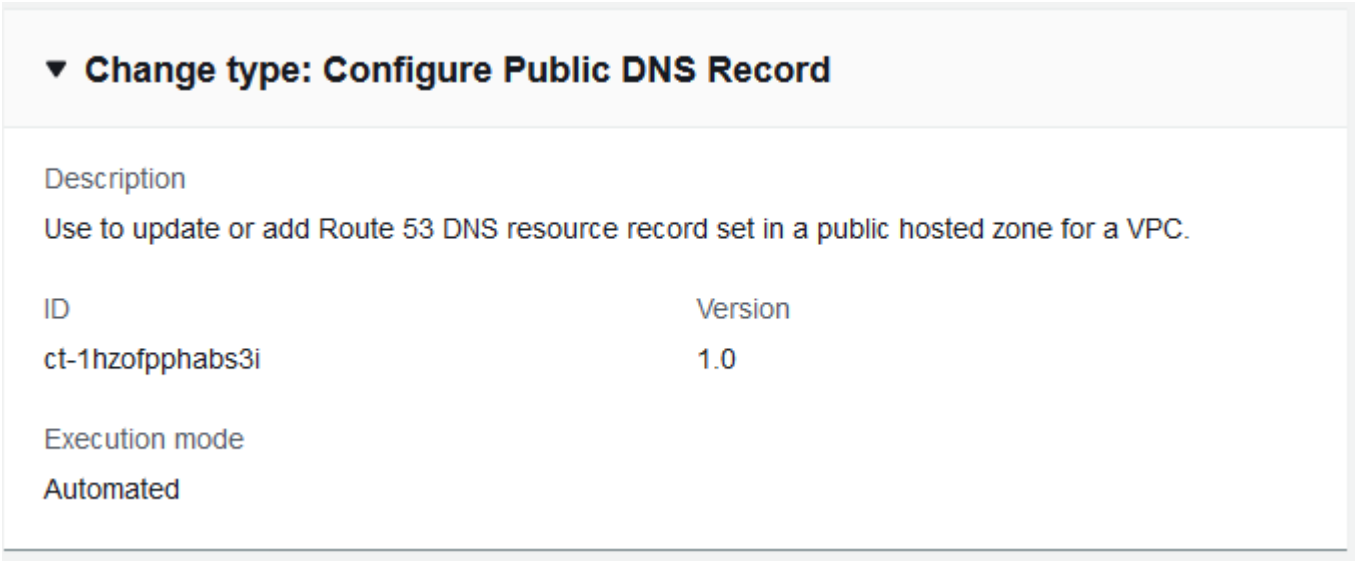
Change type ID	ct-1hzofpphabs3i
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update public DNS Route 53

Updating a Route 53 public hosted zone with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a Route 53 public hosted zone with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1hzofpphabs3i" \
--change-type-version "2.0" --title "Update Public DNS Record" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
CreateAddRoute53Resources\",\"Region\":\"ap-southeast-2\",\"Parameters\":{\"StackId
\":\"stack-6zelfurs8yojlvn1i\",\"RecordSet\":[\"{\\\"RecordSet\\\":[{\\\"Name\\\":\\
\"test15.domain.com\\\",\\\"Type\\\":\\\"A\\\",\\\"TTL\\\":600,\\\"ResourceRecords\\\":
[\\\"10.1.1.1\\\",\\\"10.1.2.2\\\"]},{\\\"Name\\\":\\\"test16.domain.com\\\",\\\"Type\\
\":\\\"CNAME\\\",\\\"TTL\\\":600,\\\"ResourceRecords\\\":[\\\"amazon.com\\\"]}]}\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateDnsPublicParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1hzofpphabs3i"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateDnsPublicParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-CreateAddRoute53Resources",
  "Region": "ap-southeast-2",
  "Parameters": {
    "StackId": "stack-6zelfurs8yojlvn1i",
    "RecordSet": [
      "{\"RecordSet\":[{\"Name\":\"test15.domain.com\",\"Type\":\"A\",\"TTL\":600,
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test16.domain.com\",
\"Type\":\"CNAME\",\"TTL\":600,\"ResourceRecords\":[\"amazon.com\"]}]}"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it UpdateDnsPublicRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateDnsPublicRfc.json
```

4. Modify and save the UpdateDnsPublicRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-1hzofpphabs3i",
  "ChangeTypeVersion": "2.0",
  "Title": "Update Public Hosted Zone"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateDnsPublicRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateDnsPublicRfc.json --execution-
parameters file://UpdateDnsPublicParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can add up to 50 tags, but to do so you must enable the **Additional configuration** view.

To learn more, see Working with Public Hosted Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1hzofpphabs3i.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateAddRoute53Resources",
  "Region" : "us-east-1",
  "Parameters": {
    "StackId": "stack-k9hasbgx6eh4d5mab",
    "RecordSet": [
      "{\"RecordSet\":[{\"Name\":\"test1.mydomain.com\",\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},{\"Name\":\"test3.mydomain.com
\",\"Type\":\"CNAME\",\"TTL\":\"600\",\"ResourceRecords\":[\"amazon.com\"]},
{\"Name\":\"test4.mydomain.com\",\"Type\":\"A\",\"AliasTarget\":{\"DNSName\":
\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,\"HostedZoneId
\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"weighted.mydomain.com\",\"Weight\":200,
\"SetIdentifier\":\"Example-Set-Identifier-1\",\"Type\":\"A\",\"AliasTarget\":
{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"geolocationexample.mydomain.com
\",\"SetIdentifier\":\"Example-GeoLocation-Identifier-1\",\"GeoLocation\":
{\"CountryCode\":\"US\",\"SubdivisionCode\":\"WA\"},\"Type\":\"A\",\"AliasTarget
\":{\"DNSName\":\"d1i3674zujyzy1.cloudfront.net\",\"EvaluateTargetHealth\":true,
\"HostedZoneId\":\"Z2FDTNDATAQYW2\"}},{\"Name\":\"examplelatency.mydomain.com\",
\"SetIdentifier\":\"Example-Latency-Identifier-1\",\"Region\":\"ap-southeast-2\",
```

```
\"Type\":\"A\",\"TTL\":\"600\",\"ResourceRecords\":[\"10.1.1.1\",\"10.1.2.2\"]},
{\"Name\":\"examplemultivalue.mydomain.com\",\"SetIdentifier\":\"Example-
MultiValue-Identifier-1\",\"MultiValueAnswer\":true,\"Type\":\"A\",\"TTL\":\"600\",
\"ResourceRecords\":[\"10.1.1.1\"]}]}"
    ]
  }
}
```

EBS Snapshot | Archive

Archive Elastic Block Store (EBS) snapshots. The maximum number of EBS snapshots that can be archived concurrently depends on the 'In-progress snapshot archives per account' AWS Service Quota. Snapshots that are in the 'completed' state, storage tier is 'standard', or belonging to the current owner account, can be archived. Snapshots created by the AWS Backup service, used by AMIs, or shared with other accounts, cannot be archived. If you specify snapshots that are invalid, or the archival in-progress quota limit is reached, the RFC fails.

Full classification: Management | Advanced stack components | EBS snapshot | Archive

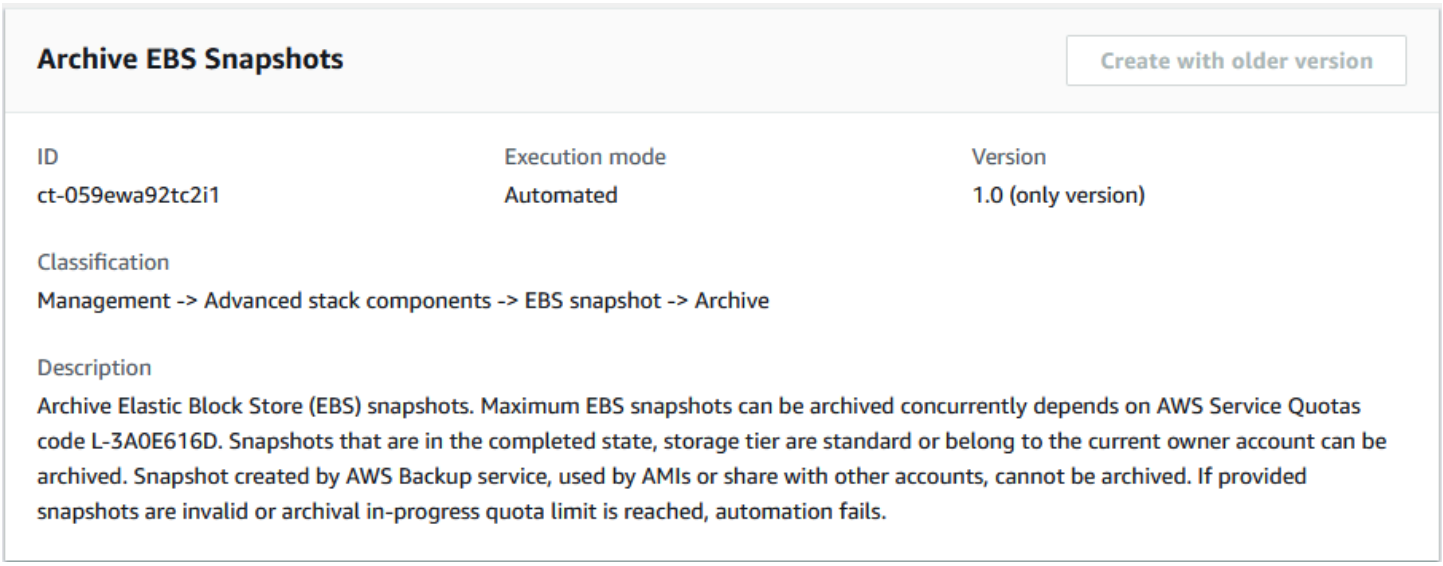
Change Type Details

Change type ID	ct-059ewa92tc2i1
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Archive EBS snapshot

Archiving EBS Snapshots with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Archiving EBS Snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-059ewa92tc2i1" --change-type-version
 "1.0" --title "Archive an EBS Snapshot" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-ArchiveEBSSnapshot\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"SnapshotId\": [\"snap-1234567890abcdef0\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ArchiveEbsSnpshtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-059ewa92tc2i1"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ArchiveEbsSnpshtParams.json
   ```

2. Modify and save the ArchiveEbsSnpshtParams file. For example, you can replace the contents with something like this:

   ```
   >{
     "DocumentName": "AWSManagedServices-ArchiveEBSSnapshot",
     "Region": "us-east-1",
     "Parameters": {
       "SnapshotId": [
         "snap-1234567890abcdef0"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it ArchiveEbsSnpshtRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ArchiveEbsSnpshtRfc.json
   ```

4. Modify and save the ArchiveEbsSnpshtRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0wspy4o646g9p",
"Title":                "Archive an EBS Snapshot"
}
```

5. Create the RFC, specifying the ArchiveEbsSnpshtRfc file and the ArchiveEbsSnpshtParams file:

```
aws amscm create-rfc --cli-input-json file://ArchiveEbsSnpshtRfc.json  --execution-
parameters file://ArchiveEbsSnpshtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS snapshots, see Amazon EBS Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-059ewa92tc2i1.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-ArchiveEBSSnapshots",
  "Region": "us-east-1",
  "Parameters": {
    "SnapshotIds": [
      "snap-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ArchiveEBSSnapshots",
  "Region": "us-east-1",
  "Parameters": {
    "SnapshotIds": [
```

```
      "snap-1234567890abcdef0"
    ]
  }
}
```

EBS Snapshot | Delete

Delete Elastic Block Store (EBS) snapshots. Because deleted snapshots cannot be restored, we recommend scheduling this RFC to provide a time period in which you could cancel the operation, if needed. At least one parameter must be specified. Note: If more than one parameter is used, only snapshots matching all used parameters are deleted. Snapshots created by AWS Backup service, used by AMIs, and snapshots created less than 60 days ago, cannot be deleted. If one or more snapshots cannot be deleted, execution fails. Up to 1000 snapshots can be deleted in one execution.

Full classification: Management | Advanced stack components | EBS snapshot | Delete

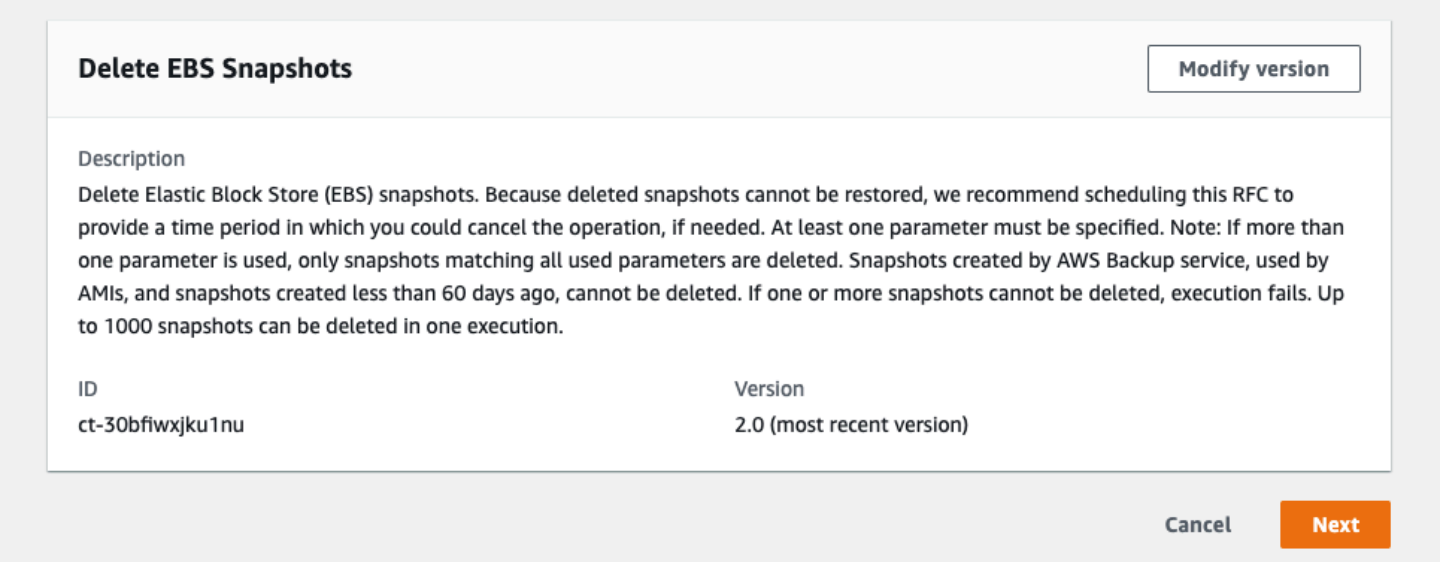
Change Type Details

Change type ID	ct-30bfiwxjku1nu
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete EBS snapshot

Deleting EBS snapshots with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting EBS snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note that the DocumentName in version 1 is AWSManagedServices-DeleteEBSSnapshot; in version 2 it is AWSManagedServices-DeleteEBSSnapshots. These examples are for version 2.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

With only `SnapshotIds` specified:

```
aws amscm create-rfc --change-type-id "ct-30bfiwxjku1nu" --change-type-version
 "2.0" --title "Delete EBS snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-DeleteEBSSnapshots\",\"Region\":\"us-east-1\",\"Confirmation\":
\"delete permanently\",\"Parameters\":{\"SnapshotIds\":[\"snap-0123456789abcdef0\",
\"snap-0123456789abcdef1\"]}}"
```

With up to 1000 snapshots listed in an S3 file specified:

```
aws amscm create-rfc --change-type-id "ct-30bfiwxjku1nu" --change-type-version
 "2.0" --title "Delete EBS Snapshots" --execution-parameters ""{\"DocumentName\":
\"AWSManagedServices-DeleteEBSSnapshots\",\"Region\":\"us-east-1\",\"Confirmation\":
\"delete permanently\",\"Parameters\":{\"SnapshotIdCsvUrl\":[\"PRE-SIGNED_S3_URL\"]}}"
```

Delete up to 1000 snapshots older than 2020-01-31 and tagged with Delete:True:

```
aws amscm create-rfc --change-type-id "ct-30bfiwxjku1nu" --change-type-version
 "2.0" --title "Delete EBS Snapshots" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-DeleteEBSSnapshots\",\"Region\":\"us-east-1\",\"Confirmation\":
\"delete permanently\",\"Parameters\":{\"StartDate\":[\"2020-01-31\"],\"Tag\":[{\"Key
\":\"Delete\",\"Value\":\"True\"}]}}"
```

Delete up to 1000 snapshots older than 2020-01-31 for which source volumes no longer exist:

```
aws amscm create-rfc --change-type-id "ct-30bfiwxjku1nu" --change-type-version
 "2.0" --title "Delete EBS Snapshots" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-DeleteEBSSnapshots\",\"Region\":\"us-east-1\",\"Confirmation
\":\"delete permanently\",\"Parameters\":{\"StartDate\":[\"2020-01-31\"],
\"SnapshotsWithoutVolumes\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteEbsSnpshtParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-30bfiwxjku1nu"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeleteEbsSnpshtParams.json
```

2. Modify and save the DeleteEbsSnpshtParams file. For example, you can replace the contents with something like this:

   With only `SnapshotIds` specified:

```
{
  "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
  "Region": "us-east-1",
  "Confirmation": "delete permanently",
  "Parameters" : {
    "SnapshotIds": [
      "snap-0123456789abcdef0",
      "snap-0123456789abcdef1"
    ]
  }
}
```

   With up to 1000 snapshots listed in an S3 file specified:

```
{
 "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
 "Region": "us-east-1",
 "Confirmation": "delete permanently",
 "Parameters": {
     "SnapshotIdCsvUrl": [
         "PRE-SIGNED_S3_URL"
     ]
 }
}}
```

   Delete up to 1000 snapshots older than 2020-01-31 and tagged with Delete:True:

```
{
 "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
```

```
 "Region": "us-east-1",
 "Confirmation": "delete permanently",
 "Parameters": {
      "StartDate": [
          "2020-01-31"
      ],
      "Tag": [
          {"Key":"Delete","Value":"True"}
      ]
 }
}
```

Delete up to 1000 snapshots older than 2020-01-31 for which source volumes no longer exist:

```
{
 "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
 "Region": "us-east-1",
 "Confirmation": "delete permanently",
 "Parameters": {
      "StartDate": [
          "2020-01-31"
      ],
      "SnapshotsWithoutVolumes": [
          "True"
      ]
 }
}
```

3. Output the RFC template JSON file to a file; this example names it DeleteEbsSnpshtRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteEbsSnpshtRfc.json
```

4. Modify and save the DeleteEbsSnpshtRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-30bfiwxjku1nu",
"Title":                "EBS-Snapshot-Delete-RFC"
}
```

5. Create the RFC, specifying the DeleteEbsSnpshtRfc file and the DeleteEbsSnpshtParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteEbsSnpshtRfc.json  --execution-
parameters file://DeleteEbsSnpshtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If more than one parameter is used, only snapshots matching all used parameters are deleted.
> Snapshots created less than 60 days ago cannot be deleted. To delete snapshots less than 60 days old, use the Management | Other | Other | Update change type (ct-0xdawir96cy7k) or a service request and an AMS engineer will assist you.
> Additionally, this CT can't delete snapshots used by AMIs or created by AWS Backup service.

> **Note**
>
> This change type is now at version 2.0 because new parameters were added to give more flexibility in determining which snapshots would be deleted. The DocumentName in version 1 is AWSManagedServices-DeleteEBSSnapshot; in version 2 it is AWSManagedServices-DeleteEBSSnapshots.

To learn more about Amazon EBS snapshots, see Amazon EBS Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-30bfiwxjku1nu.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
  "Region": "us-east-1",
```

```
  "Confirmation": "delete permanently",
  "Parameters" : {}
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-DeleteEBSSnapshots",
    "Region": "us-east-1",
    "Confirmation": "delete permanently",
    "Parameters": {
        "SnapshotIds": [
            "snap-01234567891234501",
            "snap-01234567891234502",
            "snap-01234567891234503",
            "snap-01234567891234504",
            "snap-01234567891234505",
            "snap-01234567891234506",
            "snap-01234567891234507",
            "snap-01234567891234508",
            "snap-01234567891234509",
            "snap-01234567891234510"
        ],
        "SnapshotIdCsvUrl": [
            "https://s3.us-east-1.amazonaws.com/my-bucket-0123456789/snapshots.txt?
X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=ABCDEFGHIJKLMNOPRSTU
%2F20200821%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20200821T000453Z&X-
Amz-Expires=600&X-Amz-SignedHeaders=host&X-Amz-Security-
Token=0123456789uX2VjEGgaCXVzLWVhc3QtMSJGMEQCICDq9VkeEyrvJsAbzTrb7QDMfFHY28C8BxgK0WQyKTzmAiA1fI
%2FNNGywqpsnm8GqkqyYfQlfzzPLhWgt9hMBHnEIkhY4sSGmYrRuwOwB%2B187y3imfCReNYrkhbR2SykMO
%2BRgFy2buoGXpWBYmWH2pT9IV2aTlKHj9hk7cdCfGfjpIfPYpdXPEoMY%2F1L8BdT94MgwpOqFvKBCpt
%2Fhy%2BG3EP6E1KWZK9Re%2BnIpTTzpKMXSM6HAlnl5JfOHWPm8DK6c4IwTPJtvlrJFSFYwYdFU3tO
%2FRQmXdVgS8H1LH3ug8tMN3y1SP0uHGub7pM4dcLqOGOTWN6%2F8cofyB33gw9pz8%2BQU6ngFQqBiQIowdj4y35%2Facx
%2FAv0yWidW3MiWr%2Bhc4sBSnol
%2FjfDoWx4g4LzAyJlaz51UGsCqlqWbxSODyslqu5jSnk0On0gRdHHCi8zSkwn4ornnFzsEuMDaigIFdvbkfF8q7eFMy8QN
%2FJxxFFh6yI9QF6H4bzIB1UzE0x%2FohCbQBZtda7Q%3D%3D&X-Amz-
Signature=01234567890fa9d3ebbf26fb5773017de2cc9bc10b50616f04d7932aad5e5473"
        ],
        "SnapshotCreationDate": [
            "2020-01-31"
        ],
        "SnapshotTag": [
            "{\"Key\":\"Delete\",\"Value\":\"True\"}"
```

```
        ],
        "SnapshotsWithoutVolumes": [
            "False"
        ]
    }
}
```

EBS Snapshot | Share

Share an Elastic Block Store (EBS) snapshot with another AMS account. If the destination account is onboarded in a different AMS Region, use change type ID ct-3lkbpansfv69k in the destination account to copy shared snapshot across regions. Only snapshots encrypted with managed KMS keys can be shared.

Full classification: Management | Advanced stack components | EBS snapshot | Share

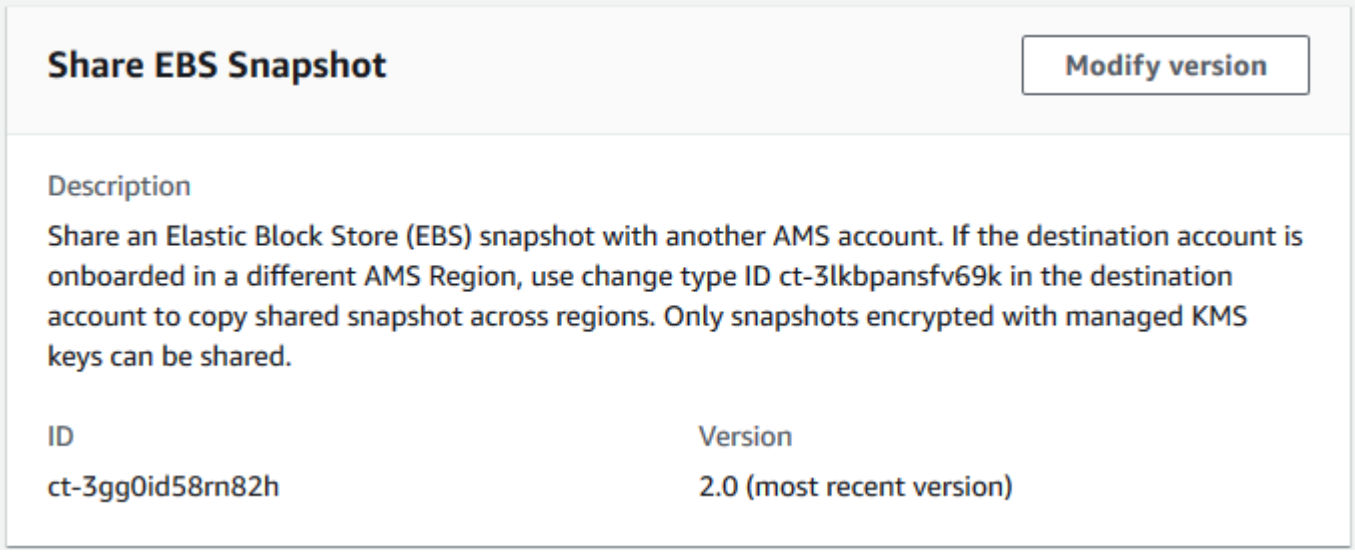
Change Type Details

Change type ID	ct-3gg0id58rn82h
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Share EBS snapshot

Sharing EBS Snapshots with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Sharing EBS Snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3gg0id58rn82h" --change-type-version
 "2.0" --title "Share EBS snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ShareEBSSnapshot\",\"Region\":\"ap-southeast-2\",\"Parameters\":
{\"AccountId\":[\"ACCOUNT_ID\"],\"SnapshotId\":[\"SNAP_ID\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ShareEbsSnpshtParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3gg0id58rn82h" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > ShareEbsSnpshtParams.json
   ```

2. Modify and save the ShareEbsSnpshtParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-ShareEBSSnapshot",
     "Region": "us-east-1",
     "Parameters": {
       "AccountId": [
         "ACCOUNT_ID"
       ],
       "SnapshotId": [
         "SNAPSHOT_ID"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it ShareEbsSnpshtRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ShareEbsSnpshtRfc.json
   ```

4. Modify and save the ShareEbsSnpshtRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-3gg0id58rn82h",
"Title":                "EBS-Share-RFC"
}
```

5. Create the RFC, specifying the ShareEbsSnpshtRfc file and the ShareEbsSnpshtParams file:

```
aws amscm create-rfc --cli-input-json file://ShareEbsSnpshtRfc.json  --execution-
parameters file://ShareEbsSnpshtParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Note

A typical use for the EBS snapshot share and copy CTs would be:

1. In account A, use the EBS snapshot share CT to share the snapshot with account B.
2. In account B, use the Copy EBS snapshot CT to copy the snapshot to the AWS Region for account B.

Important

This change type version, 2.0, limits snapshot sharing to only snapshots encrypted with managed KMS keys. Additionally, several parameters were removed, **TargetParameterName**, **Targets**, **MaxConcurrency**, and **MaxErrors**; and one new parameter was introduced, **SourceSnapshotId**.

To learn more about Amazon EBS snapshots, see Amazon EBS Snapshots.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3gg0id58rn82h.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-ShareEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SnapshotId": [
      "snap-1234567890abcdef0"
    ],
    "AccountId": [
      "012345678912"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ShareEBSSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "SnapshotId": [
      "snap-1234567890abcdef0"
    ],
    "AccountId": [
      "012345678912"
    ]
  }
}
```

EBS Volume | Attach

Attach an EBS volume to an EC2 instance. This change type provides an option that attempts to remediate drift in the CloudFormation stack where the volume is being attached, but that option, RemediateStackDrift, does not work on volumes created using the CloudFormation ingest change type (ct-36cn2avfrrj9v).

Full classification: Management | Advanced stack components | EBS Volume | Attach

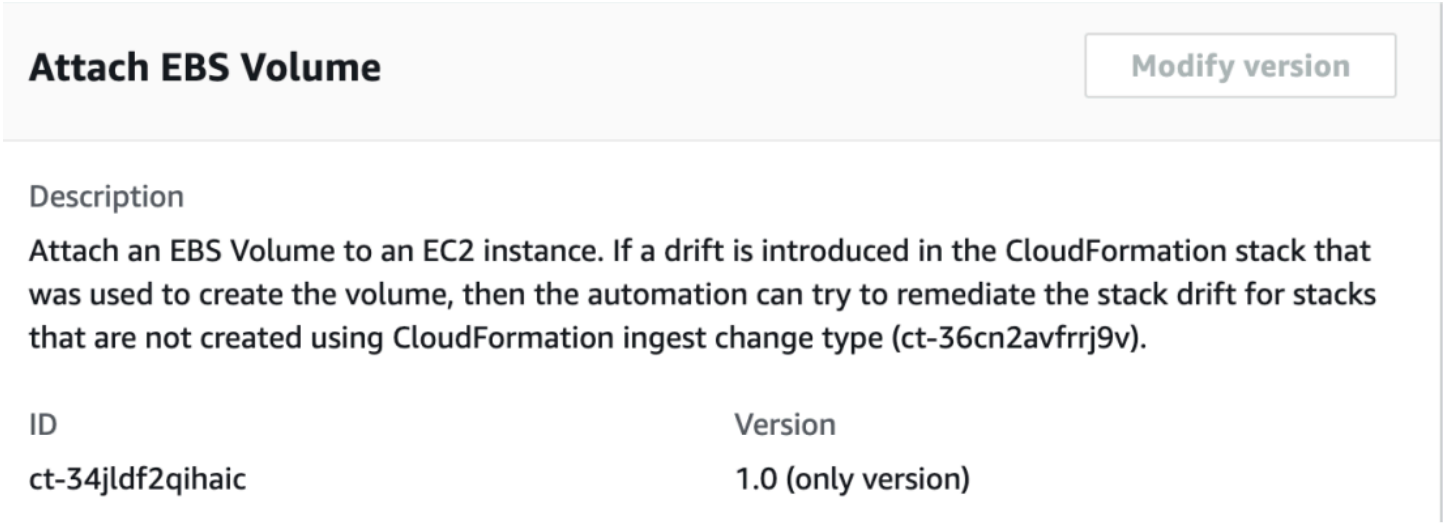
Change Type Details

Change type ID	ct-34jldf2qihaic
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Attach EBS volume

Attaching EBS Volumes with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Attaching EBS Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-34jldf2qihaic" --change-type-version
 "1.0" --title "Attach EBS Volume" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-AttachEBSVolume\",\"Region\":\"us-east-1\",\"Parameters\":
{\"VolumeId\":[\"vol-1234567890abcdef0\"],\"InstanceId\":[\"i-1234567890abcdef0\"],
\"RemediateStackDrift\":[\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AttachEbsVolParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-34jldf2qihaic" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AttachEbsVolParams.json
   ```

2. Modify and save the AttachEbsVolParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-AttachEBSVolume",
     "Region" : "us-east-1",
   ```

```
  "Parameters" : {
    "VolumeId" : [
      "vol-1234567890abcdef0"
    ],
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "RemediateStackDrift" : [
      "False"
    ]
  }
}
```

3. Output the RFC template JSON file to a file; this example names it AttachEbsVolRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AttachEbsVolRfc.json
```

4. Modify and save the AttachEbsVolRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-34jldf2qihaic",
"Title":                "EBS-Volumes-Attach-RFC"
}
```

5. Create the RFC, specifying the AttachEbsVolRfc file and the AttachEbsVolParams file:

```
aws amscm create-rfc --cli-input-json file://AttachEbsVolRfc.json  --execution-
parameters file://AttachEbsVolParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS volumes, see Amazon Elastic Block Store.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-34jldf2qihaic.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AttachEBSVolume",
  "Region" : "us-east-1",
  "Parameters" : {
    "VolumeId" : [
      "vol-1234567890abcdef0"
    ],
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "RemediateStackDrift" : [
      "False"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AttachEBSVolume",
  "Region" : "us-east-1",
  "Parameters" : {
    "VolumeId" : [
      "vol-1234567890abcdef0"
    ],
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "DeviceName" : [
      "/dev/sdf"
    ],
    "RemediateStackDrift" : [
      "False"
    ]
  }
}
```

EBS Volume | Delete

Delete Elastic Block Store (EBS) volumes in an available state. Volumes that are not attached to an instance are in an available state and can be deleted.

Full classification: Management | Advanced stack components | EBS Volume | Delete

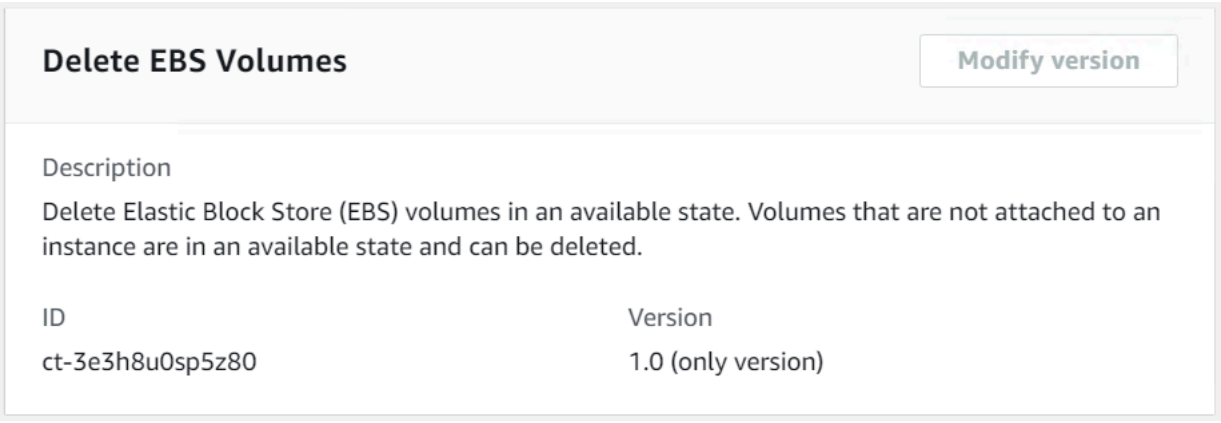
Change Type Details

Change type ID	ct-3e3h8u0sp5z80
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete EBS volume

Deleting EBS Volumes with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting EBS Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3e3h8u0sp5z80" --change-type-version
 "2.0" --title "Delete Ebs Volumes" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-DeleteEBSVolumesV2\",\"Region\":\"us-east-1\",\"Parameters\":
{\"VolumeIds\":[\"vol-01234567891234501\",\"vol-01234567891234502\"],\"CreateBackup\":
\"true\", \"DeleteStackVolume\":\"true\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteEbsVolParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3e3h8u0sp5z80" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteEbsVolParams.json
   ```

2. Modify and save the DeleteEbsVolParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-DeleteEBSVolumes",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeIds": [
      "vol-01234567891234501",
      "vol-01234567891234502"
    ],
    "CreateBackup": [
      true
    ],
    "DeleteStackVolume": [
      true
    ]
  }
}
```

3. Output the RFC template JSON file to a file; this example names it DeleteEbsVolRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteEbsVolRfc.json
```

4. Modify and save the DeleteEbsVolRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-3e3h8u0sp5z80",
"Title":                "EBS-Volumes-Delete-RFC"
}
```

5. Create the RFC, specifying the DeleteEbsVolRfc file and the DeleteEbsVolParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteEbsVolRfc.json  --execution-
parameters file://DeleteEbsVolParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS volumes, see Amazon Elastic Block Store.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3e3h8u0sp5z80.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteEBSVolumesV2",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeIds": [
      "vol-01234567891234501",
      "vol-01234567891234502",
      "vol-01234567891234503",
      "vol-01234567891234504",
      "vol-01234567891234505",
      "vol-01234567891234506",
      "vol-01234567891234507",
      "vol-01234567891234508",
      "vol-01234567891234509",
      "vol-01234567891234510"
    ],
    "CreateBackup": true,
    "DeleteStackVolume": true
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteEBSVolumesV2",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeIds": [
      "vol-01234567891234501",
      "vol-01234567891234502",
      "vol-01234567891234503",
      "vol-01234567891234504",
      "vol-01234567891234505",
      "vol-01234567891234506",
      "vol-01234567891234507",
      "vol-01234567891234508",
```

```
      "vol-01234567891234509",
      "vol-01234567891234510",
      "vol-01234567891234511",
      "vol-01234567891234512",
      "vol-01234567891234513",
      "vol-01234567891234514",
      "vol-01234567891234515",
      "vol-01234567891234516",
      "vol-01234567891234517",
      "vol-01234567891234518",
      "vol-01234567891234519",
      "vol-01234567891234520"
    ],
    "CreateBackup": true,
    "DeleteStackVolume": true
  }
}
```

EBS Volume | Detach

Detach an EBS volume from an EC2 instance. This change type provides an option that attempts to remediate drift in the CloudFormation stack where the volume is being detached, but that option, RemediateStackDrift, does not work on volumes created using the CloudFormation ingest change type (ct-36cn2avfrrj9v).

Full classification: Management | Advanced stack components | EBS Volume | Detach

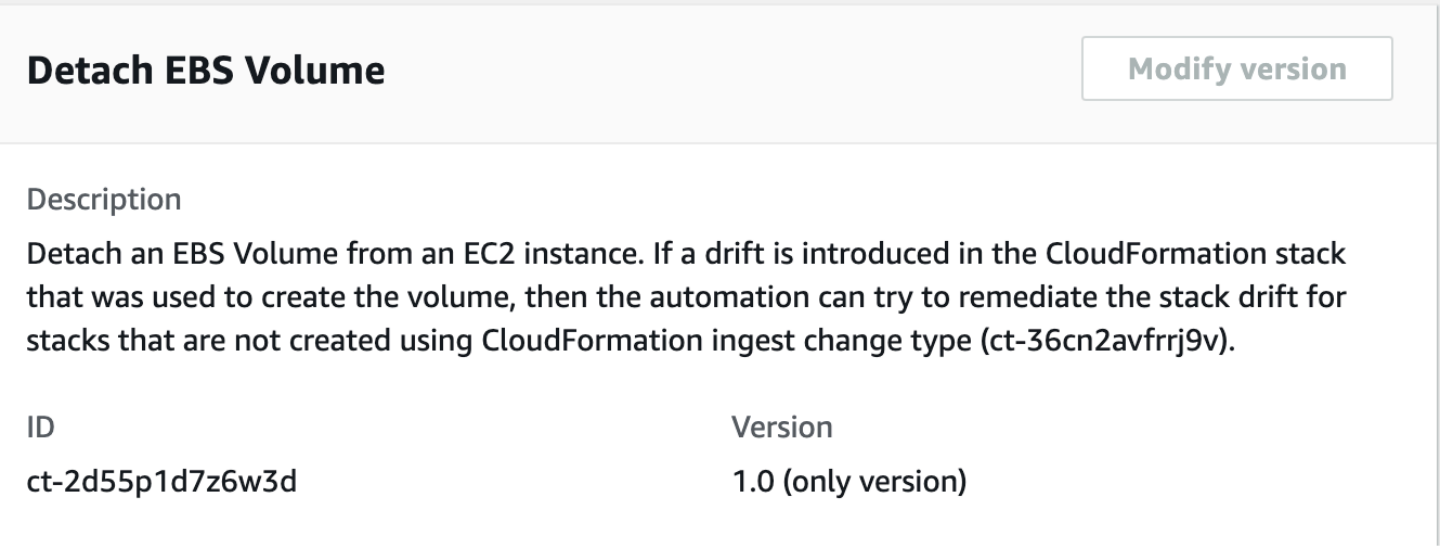
Change Type Details

Change type ID	ct-2d55p1d7z6w3d
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Detach EBS volume

Detaching EBS Volumes with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Detaching EBS Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2d55p1d7z6w3d" --change-type-version
 "1.0" --title "Detach EBS Volume" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-DetachEBSVolume\",\"Region\":\"us-east-1\",\"Parameters\":
{\"VolumeId\":[\"vol-1234567890abcdef0\"],\"RemediateStackDrift\":[\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DetachEbsVolParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2d55p1d7z6w3d" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DetachEbsVolParams.json
   ```

2. Modify and save the DetachEbsVolParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-DetachEBSVolume",
     "Region" : "us-east-1",
     "Parameters" : {
       "VolumeId" : [
         "vol-1234567890abcdef0"
       ],
       "RemediateStackDrift" : [
         "False"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it DetachEbsVolRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DetachEbsVolRfc.json
   ```

4. Modify and save the DetachEbsVolRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2d55p1d7z6w3d",
"Title":                "EBS-Volumes-Detach-RFC"
}
```

5. Create the RFC, specifying the DetachEbsVolRfc file and the DetachEbsVolParams file:

```
aws amscm create-rfc --cli-input-json file://DetachEbsVolRfc.json  --execution-
parameters file://DetachEbsVolParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS volumes, see Amazon Elastic Block Store.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2d55p1d7z6w3d.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-DetachEBSVolume",
  "Region" : "us-east-1",
  "Parameters" : {
    "VolumeId" : [
      "vol-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DetachEBSVolume",
  "Region" : "us-east-1",
```

```
  "Parameters" : {
    "VolumeId" : [
      "vol-1234567890abcdef0"
    ],
    "RemediateStackDrift" : [
      "False"
    ]
  }
}
```

EBS Volume | Encrypt EBS By Default

Set Amazon Elastic Block Store (EBS) to enforce the encryption. After you enable encryption by default, the EBS volumes that you create and snapshot copies are always encrypted, either using the KMS key configured as default for EBS encryption or the key that you specified when you created each volume.

Full classification: Management | Advanced stack components | EBS Volume | Encrypt EBS by default

Change Type Details

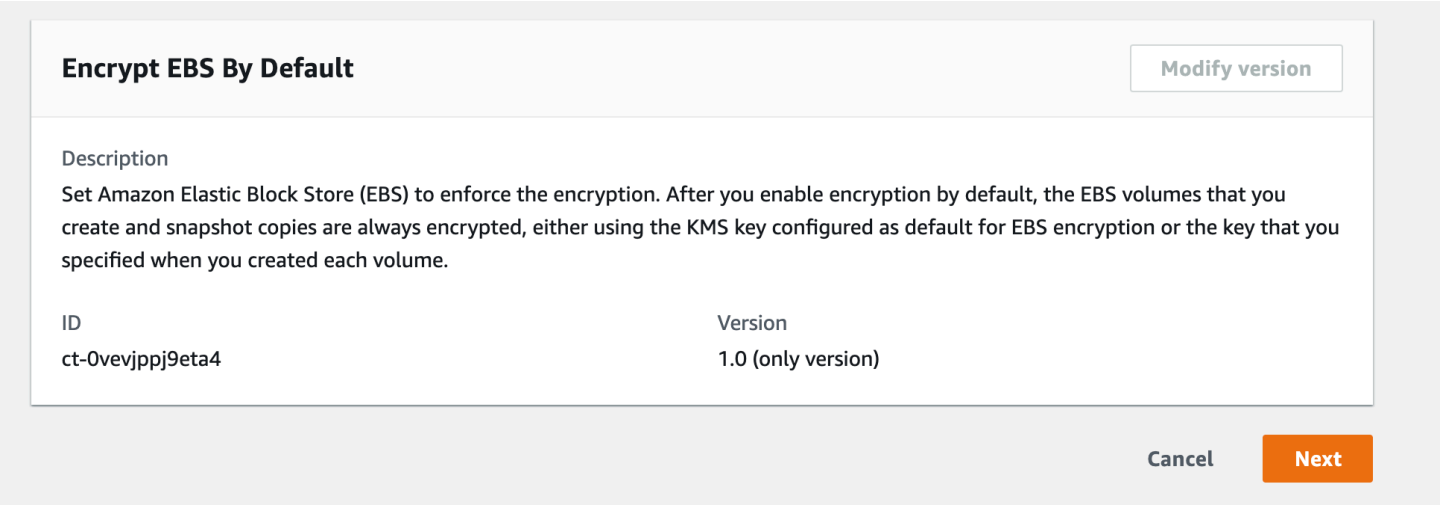
Change type ID	ct-0vevjppj9eta4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Encrypt EBS volumes by default

Set default EBS volume encryption for EC2 instances with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Set default EBS volume encryption for EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0vevjppj9eta4" --change-type-version
 "1.0" --title "Encrypt EBS by default" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-EncryptEBSByDefault\",\"Region\":\"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it EncryptEbsByDefaultParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0vevjppj9eta4"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EncryptEbsByDefaultParams.json
   ```

2. Modify and save the EncryptEbsByDefaultParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-EncryptEBSByDefault",
     "Region": "us-east-1"
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it EncryptEbsByDefaultRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EncryptEbsByDefaultRfc.json
   ```

4. Modify and save the EncryptEbsByDefaultRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-0vevjppj9eta4",
     "Title": "Encrypt EBS by default"
   }
   ```

5. Create the RFC, specifying the EncryptEbsByDefaultRfc file and the EncryptEbsByDefaultParams file:

   ```
   aws amscm create-rfc --cli-input-json file://EncryptEbsByDefaultRfc.json  --
   execution-parameters file://EncryptEbsByDefaultParams.json
   ```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> After you enable encryption by default, the EBS volumes that you create are always encrypted, either using the default AWS KMS key or the KMS key that you specified when you created each volume.
> This CT can only run in Networking and Shared Services multi-account landing zone accounts.

To learn more about doing this, see enable-ebs-encryption-by-default.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0vevjppj9eta4.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-EncryptEBSByDefault",
  "Region" : "us-east-1"
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-EncryptEBSByDefault",
  "Region" : "us-east-1"
}
```

EBS Volume | Modify

Modify EBS Volumes that are not attached to an EC2 instance in an Auto Scaling group. If you resize the volume, then you may need to extend the OS file system on the volume to use any newly

allocated space. If a drift is introduced in the CloudFormation stack that was used to create the volume, then the automation can try to remediate the stack drift for stacks that are not created using CloudFormation ingest change type (ct-36cn2avfrrj9v).

Full classification: Management | Advanced stack components | EBS Volume | Modify

Change Type Details

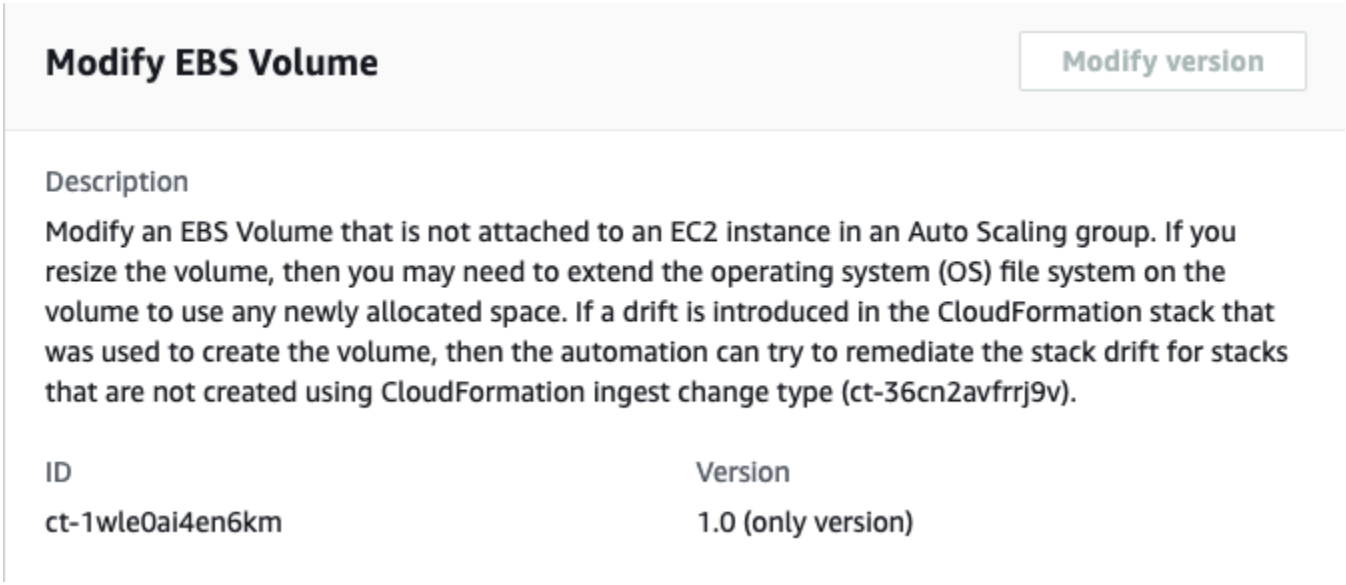
Change type ID	ct-1wle0ai4en6km
Current version	2.0
Expected execution duration	280 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Modify EBS volume

Modifying an EBS Volume with the Console

Screenshot of this change type, in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Modifying an EBS Volume with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1wle0ai4en6km" --change-type-version
 "2.0" --title "Modify EBS Volume" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ModifyEBSVolumes\",\"Region\":\"us-east-1\",\"Parameters
\":{\"VolumeIds\":[\"vol-1234567890abcdef1\",\"vol-1234567890abcdef2\",
\"vol-1234567890abcdef3\",\"vol-1234567890abcdef4\",\"vol-1234567890abcdef5\"],
\"CreateSnapshot\":[\"False\"],\"VolumeType\":[\"gp3\"],\"VolumeSize\":[\"40\"],\"Iops
\":[\"3000\"],\"Throughput\":[\"200\"],\"RemediateStackDrift\":[\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it ModifyEBSVolumeParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1wle0ai4en6km"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 ModifyEBSVolumeParams.json
```

2. Modify and save the ModifyEBSVolumeParams file.

```
    {
  "DocumentName" : "AWSManagedServices-ModifyEBSVolumes",
  "Region" : "us-east-1",
  "Parameters" : {
    "VolumeIds" : [
            "vol-1234567890abcdef1",
            "vol-1234567890abcdef2",
            "vol-1234567890abcdef3",
            "vol-1234567890abcdef4",
            "vol-1234567890abcdef5"
        ],
    "CreateSnapshot" : [
      "False"
    ],
    "VolumeType" : [
      "gp3"
    ],
    "VolumeSize" : [
      "40"
    ],
    "Iops" : [
      "3000"
    ],
    "Throughput" : [
      "200"
    ],
    "RemediateStackDrift" : [
      "False"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ModifyEBSVolumeRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ModifyEBSVolumeRfc.json
```

4. Modify and save the ModifyEBSVolumeRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-1wle0ai4en6km",
  "Title": "Modify EBS Volume"
}
```

5. Create the RFC, specifying the ModifyEBSVolumeRfc file and the ModifyEBSVolumeParams file:

```
aws amscm create-rfc --cli-input-json file://ModifyEBSVolumeRfc.json --execution-
parameters file://ModifyEBSVolumeParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EBS, see Amazon Elastic Block Store (EBS).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1wle0ai4en6km.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-ModifyEBSVolumes",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeIds": [
      "vol-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ModifyEBSVolumes",
  "Region": "us-east-1",
  "Parameters": {
    "VolumeIds": [
      "vol-01234567891234501",
      "vol-01234567891234502",
      "vol-01234567891234503",
      "vol-01234567891234504",
      "vol-01234567891234505",
      "vol-01234567891234506",
      "vol-01234567891234507",
      "vol-01234567891234508",
      "vol-01234567891234509",
      "vol-01234567891234510",
      "vol-01234567891234511",
      "vol-01234567891234512",
      "vol-01234567891234513",
      "vol-01234567891234514",
      "vol-01234567891234515",
      "vol-01234567891234516",
      "vol-01234567891234517",
      "vol-01234567891234518",
      "vol-01234567891234519",
      "vol-01234567891234520"
    ],
    "CreateSnapshot": [
      "False"
    ],
    "VolumeType": [
      "gp3"
    ],
    "VolumeSize": [
      "40"
    ],
    "Iops": [
      "3000"
    ],
    "Throughput": [
      "200"
    ],
    "RemediateStackDrift": [
```

```
      "False"
    ]
  }
}
```

EBS Volume | Update

Modify the properties of an existing Elastic Block Store (EBS) volume stack created using CT id ct-16xg8qguovg2w, version 1.0. No service interruption is expected during the update.

Full classification: Management | Advanced stack components | EBS Volume | Update

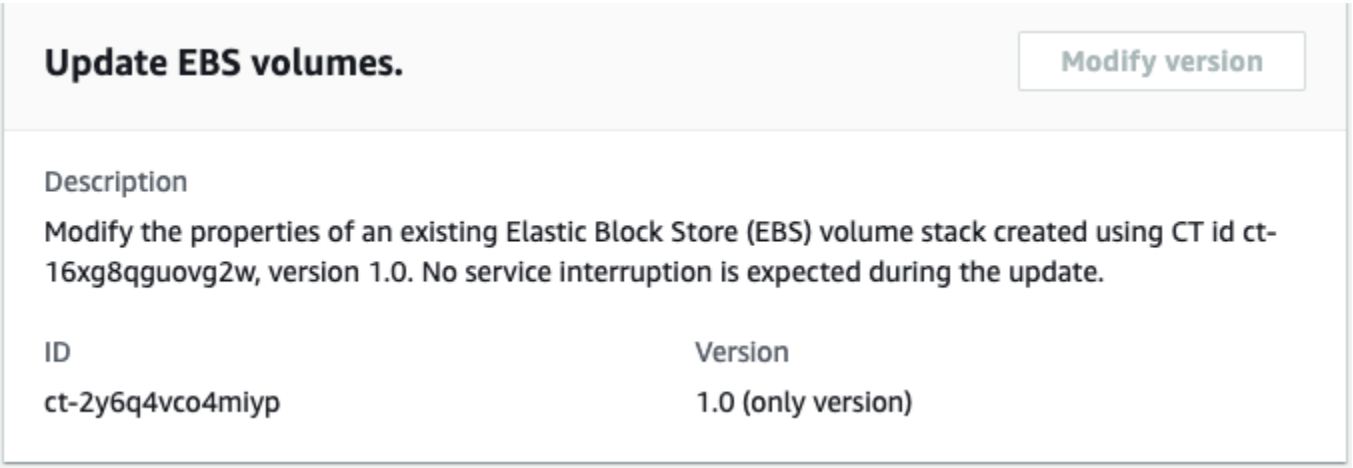
Change Type Details

Change type ID	ct-2y6q4vco4miyp
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update EBS volume

Updating EBS Volumes with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating EBS Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2y6q4vco4miyp" --change-type-
version "1.0" --title "Update EBS Volume" --execution-parameters "{\"VpcId\":
 \"vpc-0a60eb65b4EXAMPLE\",\"StackId\": \"stack-1234567890abcdef0\",\"Parameters\":
 {\"Volume1Iops\": \"3500\",\"Volume1Throughput\": \"200\",\"Volume2Type\": \"gp3\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateEbsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2y6q4vco4miyp" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateEbsParams.json
   ```

2. Modify and save the UpdateEbsParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "VpcId":   "vpc-0a60eb65b4EXAMPLE",
     "StackId": "stack-1234567890abcdef0",
     "Parameters": {
           "Volume1Iops": "3500",
            "Volume1Throughput": "200",
            "Volume2Type": "gp3"
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it UpdateEbsRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateEbsRfc.json
   ```

4. Modify and save the UpdateEbsRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   ```

```
"ChangeTypeId":         "ct-2y6q4vco4miyp",
"Title":                "Update EBS volume"
}
```

5. Create the RFC, specifying the UpdateEbsRfc file and the UpdateEbsParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateEbsRfc.json  --execution-
parameters file://UpdateEbsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For detailed information on creating RFCs, see Creating a Request for change (RFC); for an explanation of common RFC parameters, see RFC common parameters.

To learn more about Amazon EBS, see Amazon Elastic Block Store (EBS).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2y6q4vco4miyp.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "Volume1Iops": "3000",
    "Volume1Throughput": "125",
```

```
    "Volume1Size": "100",
    "Volume1Type": "gp3",
    "Volume2Iops": "3000",
    "Volume2Throughput": "125",
    "Volume2Size": "100",
    "Volume2Type": "gp3",
    "Volume3Iops": "3000",
    "Volume3Throughput": "125",
    "Volume3Size": "100",
    "Volume3Type": "gp3",
    "Volume4Iops": "3000",
    "Volume4Throughput": "125",
    "Volume4Size": "100",
    "Volume4Type": "gp3",
    "Volume5Iops": "3000",
    "Volume5Throughput": "125",
    "Volume5Size": "100",
    "Volume5Type": "gp3"
  }
}
```

EC2 Instance Stack | Associate Private IP Addresses (Review Required)

Associate one or more secondary private IP addresses to the specified network interface.

Full classification: Management | Advanced stack components | EC2 instance stack | Associate private IP addresses (review required)

Change Type Details

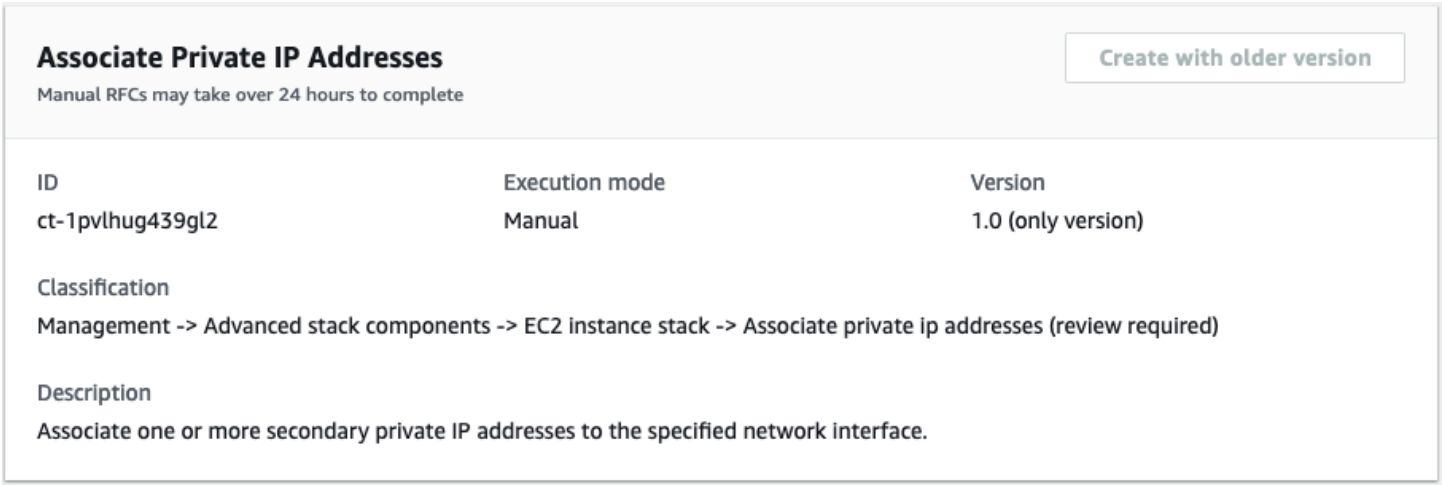
Change type ID	ct-1pvlhug439gl2
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Associate private IP addresses (review required) ct-1pvlhug439gl2

Associate private IP addresses with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating a pre-ingest instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title="Associate Private IP Addresses" --description="Associate
 Private IP Addresses" --ct-id="ct-1pvlhug439gl2" --ct-version="1.0" --input-
params="{\"NetworkInterfaceId\":\"eni-0123456789abcdef0\",\"PrivateIpAddresses\":
[\"10.0.0.82\",\"10.0.0.83\"]}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it AssociatePrivateIPAddressesParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1pvlhug439gl2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AssociatePrivateIPAddressesParams.json
   ```

2. Modify and save the AssociatePrivateIPAddressesParams file. For example, you can replace the contents with something like this:

   ```
   {
       "NetworkInterfaceId": "eni-0123456789abcdef0",
       "PrivateIpAddresses": ["10.0.0.82", "10.0.0.83"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it AssociatePrivateIPAddressesRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AssociatePrivateIPAddressesRfc.json
   ```

4. Modify and save the AssociatePrivateIPAddressesRfc.json file. For example, you can replace the contents with something like this:.

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-1pvlhug439gl2",
     "Title": "Associate Private IP Addresses"
   ```

```
}
```

5. Create the RFC, specifying the AssociatePrivateIPAddressesRfc file and the AssociatePrivateIPAddressesParams file:

```
aws amscm create-rfc --cli-input-json file://AssociatePrivateIPAddressesRfc.json
 --execution-parameters file://AssociatePrivateIPAddressesParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about Amazon EC2 IP addresses, see Amazon EC2 instance IP addressing.

If needed, see EC2 instance stack create fail.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1pvlhug439gl2.

Example: Required Parameters

```
{
    "NetworkInterfaceId": "eni-0123456789abcdef0",
    "PrivateIpAddresses": ["10.0.0.82"]
}
```

Example: All Parameters

```
{
    "NetworkInterfaceId": "eni-0123456789abcdef0",
    "PrivateIpAddresses": ["10.0.0.82", "10.0.0.83"],
    "Priority": "High"
}
```

EC2 Instance Stack | Change Hostname (Linux)

Change the hostname of an EC2 Linux instance. If no hostname is provided, then the hostname is randomized.

Full classification: Management | Advanced stack components | EC2 instance stack | Change hostname (Linux)

Change Type Details

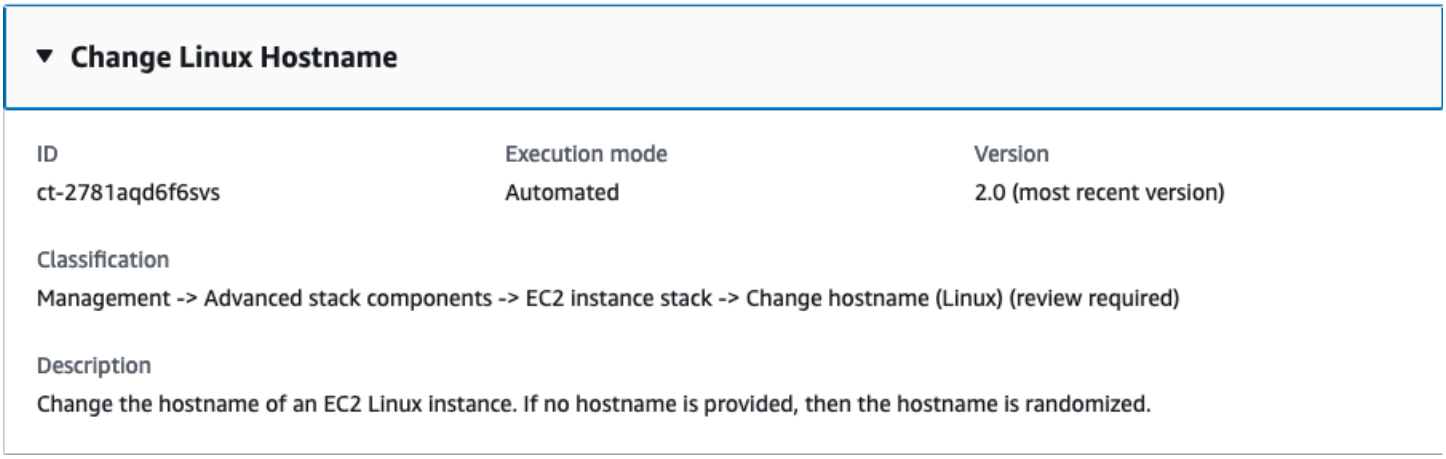
Change type ID	ct-2781aqd6f6svs
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Change hostname (Linux)

Changing the hostname for a Linux EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Changing the hostname for a Linux EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2781aqd6f6svs" --change-type-version
 "2.0" --title "Change Linux hostname" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-ChangeHostname\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"InstanceId\": [\"i-1234567890abcdef0\"],\"Hostname\": [\"01234567890abcd\"],
\"Platform\": [\"linux\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it ChangeLinuxHostnameParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2781aqd6f6svs"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ChangeLinuxHostnameParams.json
   ```

2. Modify and save the ChangeLinuxHostnameParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-ChangeHostname",
  "Region": "us-east-1",
  "Parameters": {
    "InstanceId": [ "i-1234567890abcdef0" ],
    "Hostname": [ "01234567890abcd" ],
    "Platform" : ["linux"]
  }
}}
```

3. Output the RFC template to a file in your current folder; this example names it ChangeLinuxHostnameRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ChangeLinuxHostnameRfc.json
```

4. Modify and save the ChangeLinuxHostnameRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-2781aqd6f6svs",
  "Title": "Change Linux Hostname"
}
```

5. Create the RFC, specifying the ChangeLinuxHostnameRfc file and the ChangeLinuxHostnameParams file:

```
aws amscm create-rfc --cli-input-json file://ChangeLinuxHostnameRfc.json  --
execution-parameters file://ChangeLinuxHostnameParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is at a new version, 2.0, and is now automated (version 1.0 was execution mode=manual). There are additional parameters, notably **DocumentName** and **Platform**.

To learn more about doing this, see Changing the hostname of your Amazon Linux instance.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2781aqd6f6svs.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-ChangeHostname",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "Platform" : [
      "linux"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-ChangeHostname",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "Hostname" : [
      "testhostname"
    ],
    "Platform" : [
      "linux"
    ]
  }
}
```

EC2 Instance Stack | Change Hostname (Windows)

Change the hostname of an EC2 Windows instance. Note that the instance will be rebooted.

Full classification: Management | Advanced stack components | EC2 instance stack | Change hostname (Windows)

Change Type Details

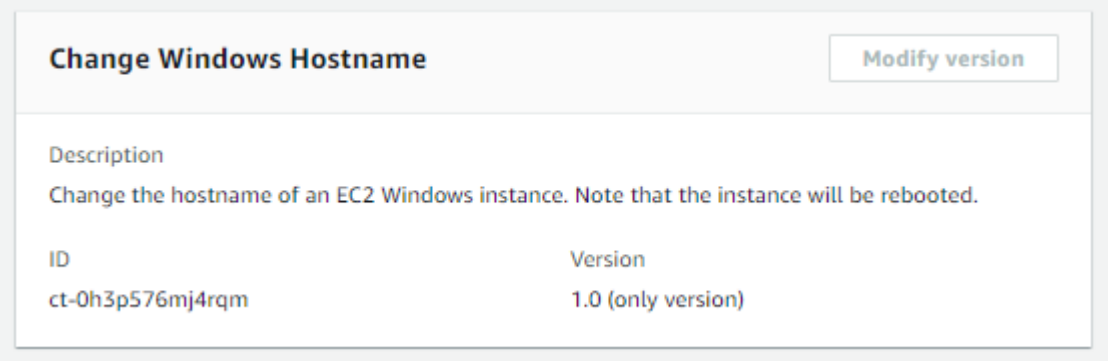
Change type ID	ct-0h3p576mj4rqm
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Change hostname (Windows)

Changing the hostname for a Windows EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Changing the hostname for a Windows EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0h3p576mj4rqm" --change-type-version
 "1.0" --title "Change Windows Hostname" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-ChangeHostname\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstanceId\":[\"i-12345678901234567\"],\"Hostname\":[\"myhost\"],\"Platform\":
[\"windows\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it ChangeWindowsHostnameParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0h3p576mj4rqm"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ChangeWindowsHostnameParams.json
   ```

2. Modify and save the ChangeWindowsHostnameParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-ChangeHostname",
   ```

```
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-12345678901234567"
    ],
    "Hostname" : [
      "myhost"
    ],
    "Platform" : [
      "windows"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ChangeWindowsHostnameRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ChangeWindowsHostnameRfc.json
```

4. Modify and save the ChangeWindowsHostnameRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-0h3p576mj4rqm",
  "Title": "Change Windows Hostname"
}
```

5. Create the RFC, specifying the ChangeWindowsHostnameRfc file and the ChangeWindowsHostnameParams file:

```
aws amscm create-rfc --cli-input-json file://ChangeWindowsHostnameRfc.json  --
execution-parameters file://ChangeWindowsHostnameParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about EC2 for Windows, see the EC2 User Guide for Windows.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0h3p576mj4rqm.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-ChangeHostname",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-12345678901234567"
    ],
    "Hostname" : [
      "testhostname"
    ],
    "Platform" : [
      "windows"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-ChangeHostname",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-12345678901234567"
    ],
    "Hostname" : [
      "testhostname"
    ],
    "Platform" : [
      "windows"
    ]
  }
}
```

EC2 Instance Stack | Change Time Zone

Change the time zone of an EC2 instance. To reboot the EC2 instance after changing the time zone, set Reboot = true.

Full classification: Management | Advanced stack components | EC2 instance stack | Change time zone

Change Type Details

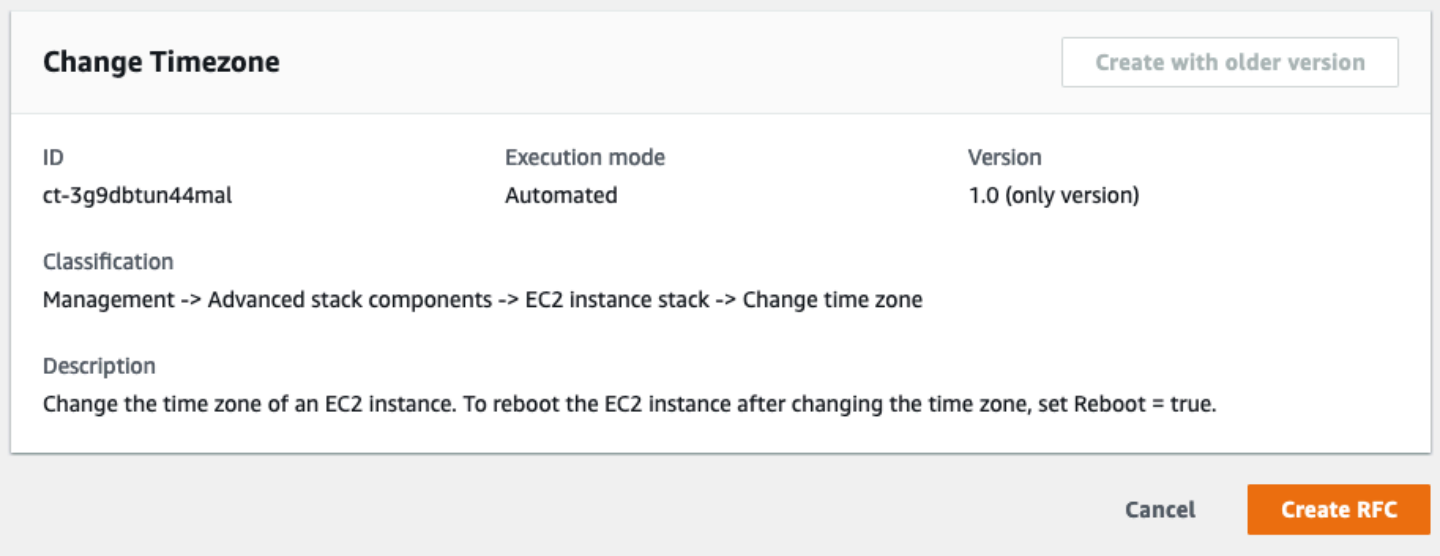
Change type ID	ct-3g9dbtun44mal
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Change time zone on instance

Changing an EC2 instance time zone with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Changing an EC2 instance time zone with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-3g9dbtun44mal" --change-type-version
 "1.0" --title "Change time zone" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-SetInstanceTimeZone\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstanceId\":\"i-1234567890abcdef0\",\"Reboot\":\"True\",\"TimeZone\":\"Australia/
Sydney (AUS Eastern Standard Time)\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it ChangeEC2TimezoneParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3g9dbtun44mal"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ChangeEC2TimezoneParams.json
   ```

2. Modify and save the ChangeEC2TimezoneParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-SetInstanceTimeZone",
       "Region": "us-east-1",
       "Parameters": {
           "InstanceId": "i-1234567890abcdef0",
           "Reboot": "True",
           "TimeZone": "Australia/Sydney (AUS Eastern Standard Time)"
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it ChangeEC2TimezoneRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ChangeEC2TimezoneRfc.json
   ```

4. Modify and save the ChangeEC2TimezoneRfc.json file. For example, you can replace the contents with something like this:.

   ```
   {
   ```

```
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3g9dbtun44mal",
  "Title": "Change EC2 Instance Time Zone"
}
```

5. Create the RFC, specifying the ChangeEC2TimezoneRfc file and the ChangeEC2TimezoneParams file:

```
aws amscm create-rfc --cli-input-json file://ChangeEC2TimezoneRfc.json  --
execution-parameters file://ChangeEC2TimezoneParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about Amazon EC2, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3g9dbtun44mal.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-SetInstanceTimeZone",
  "Region": "us-east-1",
  "Parameters": {
    "InstanceId": "i-1234567890abcdef0",
    "Reboot": "False",
    "TimeZone": "Australia/Sydney (AUS Eastern Standard Time)"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-SetInstanceTimeZone",
  "Region": "us-east-1",
```

```
  "Parameters": {
    "InstanceId": "i-1234567890abcdef0",
    "Reboot": "True",
    "TimeZone": "Australia/Sydney (AUS Eastern Standard Time)"
  }
}
```

EC2 Instance Stack | Enable Detailed Monitoring (Review Required)

Enable detailed monitoring for the specified EC2 instance. Detailed monitoring incurs a charge. EC2 detailed monitoring provides more frequent metrics, published at one-minute intervals, instead of the five-minute intervals used in Amazon EC2 basic monitoring.

Full classification: Management | Advanced stack components | EC2 instance stack | Enable detailed monitoring (review required)

Change Type Details

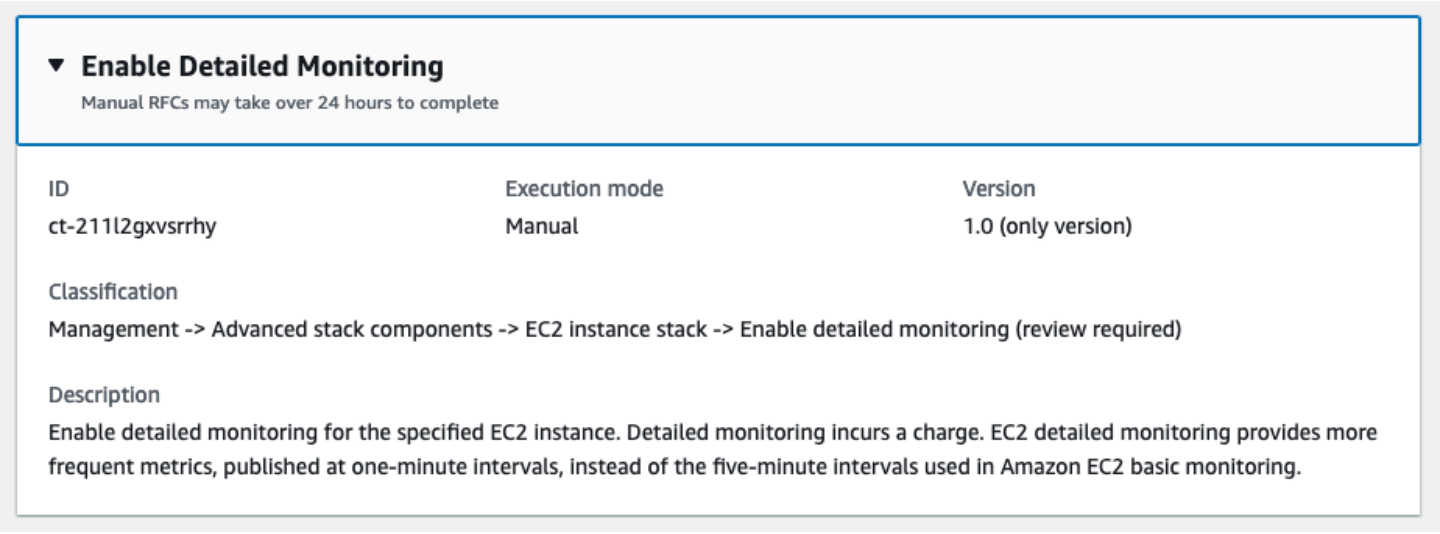
Change type ID	ct-211l2gxvsrrhy
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Enable Detailed Monitoring

Enable detailed monitoring with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enable detailed monitoring with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-211l2gxvsrrhy" --change-type-version "1.0"
 --title "Enable Detailed Monitoring" --execution-parameters "{\"InstanceIds\":
 [\"i-1234567890abcdef0\",\"i-1234567890abcdef1\"]}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it EnableDetailedMonitoringParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-211l2gxvsrrhy"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EnableDetailedMonitoringParams.json
   ```

2. Modify and save the EnableDetailedMonitoringParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
       "InstanceIds": ["i-0cc489fa851c31a21","i-0cc489fa851c31a22"]
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it EnableDetailedMonitoringRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableDetailedMonitoringRfc.json
   ```

4. Modify and save the EnableDetailedMonitoringRfc file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-211l2gxvsrrhy",
     "Title": "Enable Detailed Monitoring"
   }
   ```

5. Create the RFC, specifying the EnableDetailedMonitoringRfc file and the EnableDetailedMonitoringParams file:

```
aws amscm create-rfc --cli-input-json file://EnableDetailedMonitoringRfc.json  --
execution-parameters file://EnableDetailedMonitoringParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-211l2gxvsrrhy.

Example: Required Parameters

```
{
  "InstanceIds": ["i-1234567890abcdef0","i-1234567890abceef1"]
}
```

Example: All Parameters

```
{
  "InstanceIds": ["i-1234567890abcdef0","i-1234567890abceef1"]
}
```

EC2 Instance Stack | Encrypt Instance Volumes

Encrypt Elastic Block Store (EBS) volumes attached to an EC2 instance

Full classification: Management | Advanced stack components | EC2 instance stack | Encrypt instance volumes

Change Type Details

Change type ID	ct-0hahohe17csnc

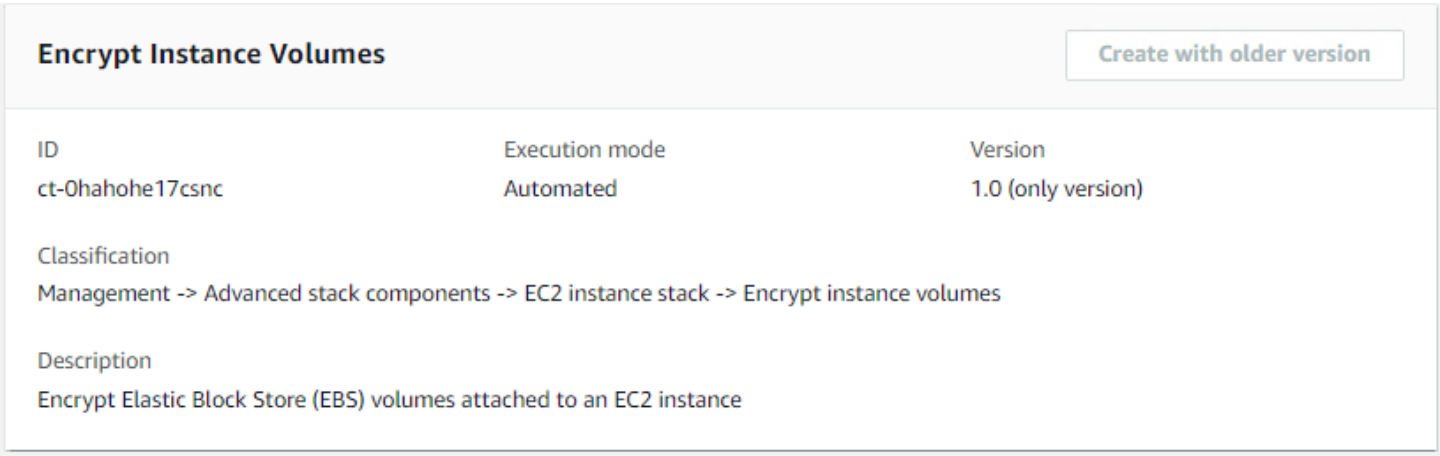
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Encrypt instance

Encrypting EC2 instances with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Encrypting EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0hahohe17csnc" --change-type-version
 "1.0" --title "AWSManagedServices-EncryptInstanceVolumes" --execution-parameters
 "{\"DocumentName\": \"AWSManagedServices-EncryptInstanceVolumes\",\"Region
\": \"us-east-1\",\"Parameters\": {\"InstanceId\": [\"i-0a458848bc91a1b7b\"],
 \"VolumeIds\": [\"vol-02f576d0e8c5c51e8\", \"vol-0090e02379b9880d9\"], \"KMSKeyId\":
 [\"7103a217-2489-481e-976c-0375efc5f606\"], \"DeleteStaleNonEncryptedSnapshotBackups
\": [\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it EncryptEC2sParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0hahohe17csnc" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > EncryptEC2sParams.json
   ```

2. Modify and save the EncryptEC2sParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-EncryptInstanceVolumes",
     "Region": "us-east-1",
     "Parameters": {
       "InstanceId": [
   ```

```
      "i-0a458848bc91a1b7b"
    ],
    "VolumeIds": [
      "vol-02f576d0e8c5c51e8",
      "vol-0090e02379b9880d9"
    ],
    "KMSKeyId": [
      "7103a217-2489-481e-976c-0375efc5f606"
    ],
    "DeleteStaleNonEncryptedSnapshotBackups": [
      "False"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it EncryptEC2sRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > EncryptEC2sRfc.json
```

4. Modify and save the EncryptEC2sRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0hahohe17csnc",
"Title":                "EC2-Encrypt-RFC"
}
```

5. Create the RFC, specifying the EncryptEC2sRfc file and the EncryptEC2sParams file:

```
aws amscm create-rfc --cli-input-json file://EncryptEC2sRfc.json  --execution-
parameters file://EncryptEC2sParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0hahohe17csnc.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-EncryptInstanceVolumes",
  "Region" : "us-east-1",
  "Parameters": {
    "InstanceId": ["i-1234567890abcdef0"],
    "VolumeIds": ["vol-1234567890abcdef0", "vol-1234567890abcdef1"],
    "KMSKeyId": ["1234abcd-12ab-34cd-56ef-1234567890ab"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-EncryptInstanceVolumes",
  "Region" : "us-east-1",
  "Parameters": {
    "InstanceId": ["i-1234567890abcdef0"],
    "VolumeIds": ["vol-1234567890abcdef0", "vol-1234567890abcdef1"],
    "KMSKeyId": ["mrk-c280f426a06049f0bd3f998242ae2f70"],
    "DeleteStaleNonEncryptedSnapshotBackups": ["False"]
  }
}
```

EC2 Instance Stack | Gather Log4j Information

Generates a report identifying Log4j2 occurrences on the specified EC2 instances. This is a best-effort report and some occurrences may go undetected from the report.

Full classification: Management | Advanced stack components | EC2 instance stack | Gather log4j information

Change Type Details

Change type ID	ct-19f40lfm5umy8

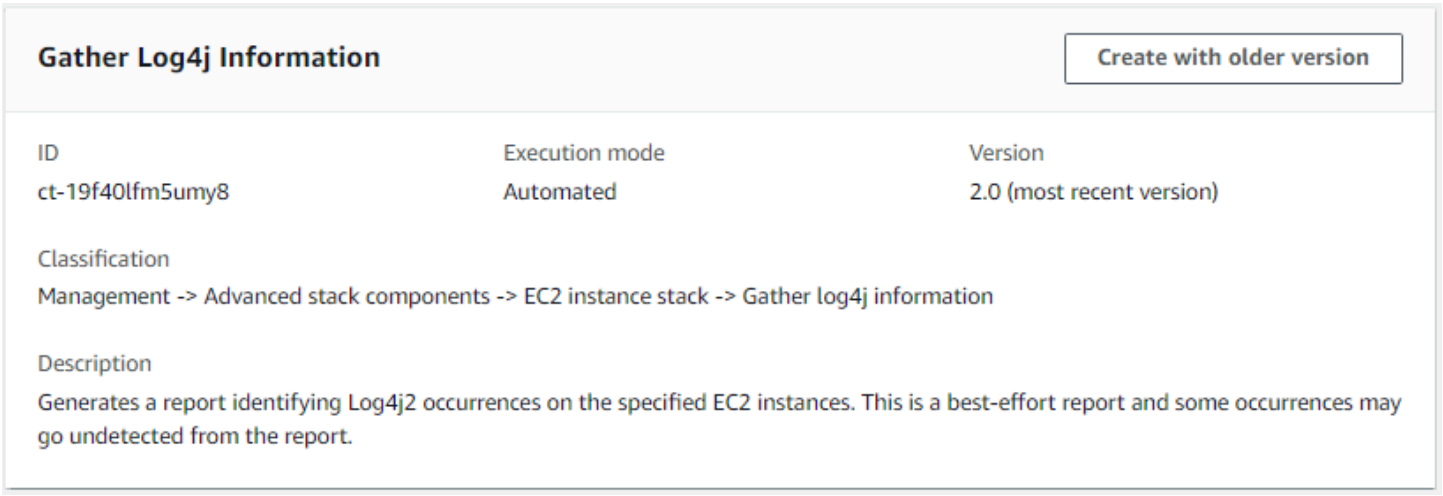
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update Other Other CTs

Gather Log4j Info on multiple EC2 instances with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Gather Log4j Info on multiple EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Version 2.0:

Scan all instances:

```
aws amscm create-rfc --change-type-id "ct-19f40lfm5umy8" --change-type-version
 "2.0" --title "Log4j Investigation" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-GatherLog4jInformation\",\"Region\":\"us-east-1\",\"Parameters\":
{\"S3Bucket\":[\"s3://BUCKET_NAME\"]},\"TargetParameterName\": \"InstanceId\",\"Targets
\": [{\"Key\": \"AWS::EC2::Instance\",\"Values\": [\"*\"]}],\"MaxConcurrency\": \"10\",
\"MaxErrors\": \"100%\"}"
```

Scan a list of instances:

```
aws amscm create-rfc --change-type-id "ct-19f40lfm5umy8" --change-type-version
 "2.0" --title "Log4j Investigation" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-GatherLog4jInformation\",\"Region\":\"us-east-1\",\"Parameters\":
{\"S3Bucket\":[\"s3://BUCKET_NAME\"]},\"TargetParameterName\": \"InstanceId\",\"Targets
\": [{\"Key\": \"ParameterValues\",\"Values\": [\"INSTANCE_ID_1\",\"INSTANCE_ID_2\",
\"INSTANCE_ID_3\",\"INSTANCE_ID_4\",\"INSTANCE_ID_5\"]}],\"MaxConcurrency\": \"10\",
\"MaxErrors\": \"100%\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it GatherLog4jInfoParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-19f40lfm5umy8"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 GatherLog4jInfoParams.json
```

2. Modify and save the GatherLog4jInfoParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   Version 2.0:

   Scan all instances:

```
{
  "DocumentName": "AWSManagedServices-GatherLog4jInformation",
  "Region": "us-east-1",
  "Parameters": {
    "S3Bucket": [
      "s3://BUCKET_NAME"
    ]
  },
  "TargetParameterName": "InstanceId",
  "Targets": [
    {
      "Key": "AWS::EC2::Instance",
      "Values": [
        "*"
      ]
    }
  ],
  "MaxConcurrency": "10",
  "MaxErrors": "100%"
}
```

   Scan a list of instances:

```
{
  "DocumentName": "AWSManagedServices-GatherLog4jInformation",
  "Region": "us-east-1",
  "Parameters": {
    "S3Bucket": [
      "s3://BUCKET_NAME"
    ]
```

```
  },
  "TargetParameterName": "InstanceId",
  "Targets": [
    {
      "Key": "ParameterValues",
      "Values": [
        "INSTANCE_ID_1",
        "INSTANCE_ID_2",
        "INSTANCE_ID_3",
        "INSTANCE_ID_4",
        "INSTANCE_ID_5"
      ]
    }
  ],
  "MaxConcurrency": "10",
  "MaxErrors": "100%"
}
```

3. Output the RFC template to a file in your current folder; this example names it GatherLog4jInfoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > GatherLog4jInfoRfc.json
```

4. Modify and save the GatherLog4jInfoRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-19f40lfm5umy8",
"Title":                "Log4j Investigation"
}
```

5. Create the RFC, specifying the GatherLog4jInfoRfc file and the GatherLog4jInfoParams file:

```
aws amscm create-rfc --cli-input-json file://GatherLog4jInfoRfc.json  --execution-
parameters file://GatherLog4jInfoParams.json
```

Tips

This change type scans the specified EC2 instance for packages containing an impacted version of the Apache Log4j Java class. This functionality produces a best-effort report, some occurrences may go undetected or mis-identified.

AWS CloudShell is a browser-based shell that makes it easy to securely manage, explore, and interact with your AWS resources. AWS CloudShell is pre-authenticated with your console credentials when you log in. Common development and operations tools are pre-installed, so no local installation or configuration is required. With AWS CloudShell, you can quickly run scripts with the AWS Command Line Interface (AWS CLI), experiment with AWS service APIs using the AWS SDKs, or use a range of other tools to be productive. You can use AWS CloudShell right from your browser at no additional cost.

> **Note**
>
> You can use the CloudShell AWS console from any other, or the closest, AWS Region where it is available, to perform the aggregation. For example, to perform the aggregation of data stored in the Virginia region, open a CloudShell in the "US East(Virginia) us-east-1" AWS Region in the AWS Console and follow the instructions given next.

The report data includes information about Java Archives (JAR Files), found within the specified environment that contain the vulnerable JndiLookup class. AMS recommends upgrading impacted libraries to the latest available version, which can be downloaded directly from Apache at Download Apache Log4j 2. Additionally, we scan for Web Application Resource (WAR), Enterprise Archive (EAR), Jupiter Encrypted XML (JPI), Hemera Technologies (HPI), and ZIP files.

To aggregate all the generated CSV files and build a single report with AWS CloudShell:

1. From any page or AWS Region in the AWS Management Console, open the AWS CloudShell to run the script shown next. Ensure that you are logged into the AWS Management Console with the AWSManagedServicesReadOnlyRole role.

```
# Specify the S3 bucket and AWS region that contains the individual CSV files:
BUCKET_NAME="YOUR BUCKET HERE"
BUCKET_REGION="THE BUCKET REGION HERE"

# Aggregate the CSV files:
mkdir -p log4j-report
```

```
aws s3 cp s3://$BUCKET_NAME/ams/log4j-scan/ ./log4j-report --recursive --include
 "*.csv"
echo "aws_account_id,region,scan_time,instance_id,scan_type,location" > log4j-
report/report.csv
for i in `find log4j-report -type f \( -iname "*.csv" ! -iname "report.csv" \)`; do
 awk 'FNR > 1' $i >> log4j-report/report.csv; done

# Upload the report to the same S3 bucket:
file_name="report_$(date -d "today" +"%Y%m%d%H%M").csv"
aws s3 cp log4j-report/report.csv s3://$BUCKET_NAME/ams/log4j-reports/$file_name

# Open the following URL and select \"Download\" to download the report:
echo "Report uploaded to: https://s3.console.aws.amazon.com/s3/object/$BUCKET_NAME?
region=$BUCKET_REGION&prefix=ams/log4j-reports/$file_name"
```

The script outputs the S3 URL to download the report from.

2. Copy and open the URL and then choose Download

Single-Account Landing Zone: Using the report

If you are working in a single-account landing zone, the AWS CloudShell service is not available. However, you can still leverage the AWS CLI to perform the necessary steps. Follow this documentation, How do I grant my Active Directory users access to the API or AWS CLI with AD FS?, to configure CLI API Access through Active Directory Federation Services (ADFS) using IAM Roles. For Non-ADFS identity provider (IDP) implementations, visit How to Implement a General Solution for Federated API/CLI Access Using SAML 2.0. Using the above options, obtain CLI Credentials for the desired role, the default recommended role is the `Customer_ReadOnly_Role`. Then execute the script in Step 1 to generate the required CSV report.

How to read the report

The report contains the following columns:

- **scan_time**: The time at which the instance scan was performed
- **instance_id**: The EC2 instance ID
- **scan_type**: The type of scan that was performed. For example, if the scan looked at in memory information, the scan_type will be MEMORY. If the filesystem was checked, the scan_type will be FILESYSTEM
- **location**: The path to the match

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-19f40lfm5umy8.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-GatherLog4jInformation",
  "Region": "us-east-1",
  "Parameters": {
    "S3Bucket": [
      "s3://test"
    ]
  },
  "TargetParameterName": "InstanceId",
  "Targets": [
    {
      "Key": "ParameterValues",
      "Values": [
        "i-1234567890abcdef0",
        "i-1234567890abcdef1",
        "i-1234567890abcdef2",
        "i-1234567890abcdef3",
        "i-1234567890abcdef4"
      ]
    }
  ],
  "MaxConcurrency": "10",
  "MaxErrors": "100%"
}
```

EC2 Instance Stack | Reboot

Use to reboot an EC2 instance.

Full classification: Management | Advanced stack components | EC2 instance stack | Reboot

Change Type Details

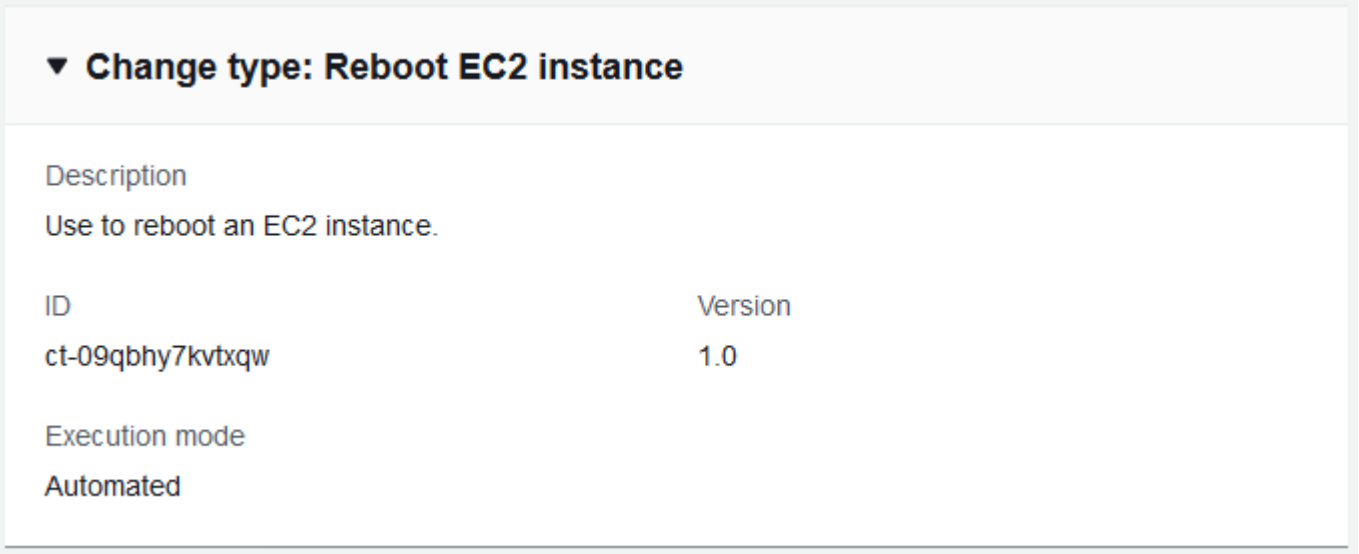
Change type ID	ct-09qbhy7kvtxqw
Current version	1.0
Expected execution duration	30 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Reboot instance

Rebooting an EC2 istance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Rebooting an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-02u0hoaa9grat" --change-type-version "1.0" --
title "Reboot My EC2" --execution-parameters "{\"InstanceId\":\"INSTANCE_ID\"}"
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder. This example names it RebootEC2Rfc.json. Note that since there is only one execution parameter for stopping (rebooting, or starting) an instance, the execution parameter can be in the schema JSON file itself and there is no need to create a separate execution parameters JSON file.

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopInstanceRfc.json
   ```

2. Modify and save the RebootEC2Rfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-09qbhy7kvtxqw",
   ```

```
"Title":                "Reboot-My-EC2-RFC",
"TimeoutInMinutes":     60,
"ExecutionParameters":  "{
        \"InstanceId\":\"INSTANCE_ID\"
    }"
}
```

3. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://RebootEC2Rfc.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about EC2, see the EC2 Documentation for your operation system.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-09qbhy7kvtxqw.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

EC2 Instance Stack | Replace Instance Profile

Replace the instance profile of an EC2 instance that is not part of an Auto Scaling group. This change may result in CloudFormation drift for any stacks that have this resource.

Full classification: Management | Advanced stack components | EC2 instance stack | Replace instance profile

Change Type Details

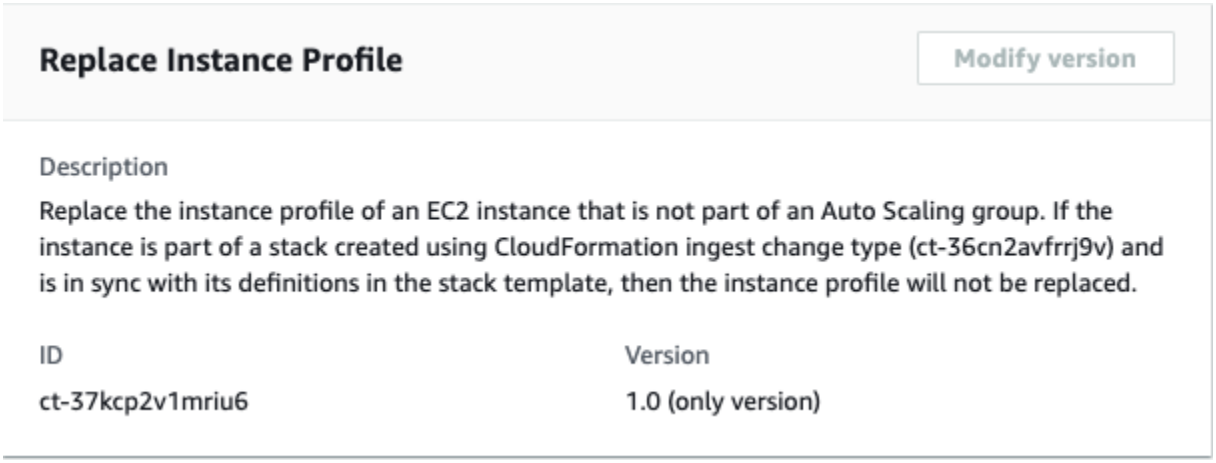
Change type ID	ct-37kcp2v1mriu6
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Replace instance profile

Replacing an EC2 instance profile using the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Replacing an EC2 instance profile using the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-37kcp2v1mriu6" --change-type-version
 "1.0" --title "Replace Instance Profile" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ReplaceInstanceProfile\",\"Region\":\"us-east-1\",\"Parameters
\":{\"InstanceId\":[\"i-12345678901234567\"],\"InstanceProfile\":[\"customer-test-
profile\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it ReplaceEC2InstanceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-37kcp2v1mriu6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ReplaceEC2InstanceParams.json
   ```

2. Modify and save the ReplaceEC2InstanceParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-ReplaceInstanceProfile",
     "Region": "us-east-1",
   ```

```
  "Parameters": {
    "InstanceId": [
      "i-12345678901234567"
    ],
    "InstanceProfile": [
      "customer-test-profile"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ReplaceEC2InstanceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ReplaceEC2InstanceRfc.json
```

4. Modify and save the ReplaceEC2InstanceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-37kcp2v1mriu6",
"Title":                "Replace Instance Profile"
}
```

5. Create the RFC, specifying the ReplaceEC2InstanceRfc file and the ReplaceEC2InstanceParams file:

```
aws amscm create-rfc --cli-input-json file://ReplaceEC2InstanceRfc.json  --
execution-parameters file://ReplaceEC2InstanceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information on instance profiles, see Using instance profiles.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-37kcp2v1mriu6.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-ReplaceInstanceProfileV2",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "InstanceProfile" : [
      "customer-test-profile"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-ReplaceInstanceProfileV2",
  "Region" : "us-east-1",
  "Parameters" : {
    "InstanceId" : [
      "i-1234567890abcdef0"
    ],
    "InstanceProfile" : [
      "customer-test-profile"
    ]
  }
}
```

EC2 Instance Stack | Resize

Resize an existing EC2 instance in your account. The state of the instance can be either 'running' or 'stopped'. If 'running', the instance is stopped during the resize operation and returned to the initial state after the resizing is complete. Before resizing the instance, ensure that the instance's root volume is not an instance store volume. We highly recommended rigorous load and performance testing before, and after, making instance type changes, and that you also consider the pricing changes that result when instances are resized. Please be aware that this change may result in CloudFormation drift for any stacks that have this resource.

Full classification: Management | Advanced stack components | EC2 instance stack | Resize

Change Type Details

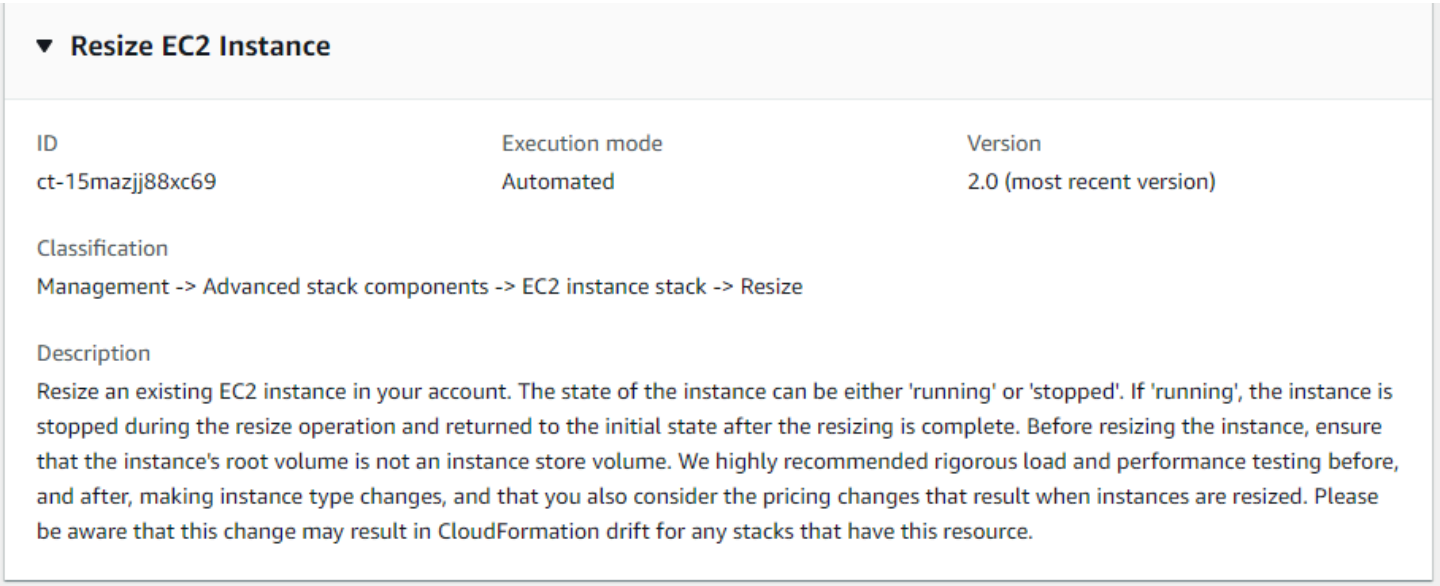
Change type ID	ct-15mazjj88xc69
Current version	2.0
Expected execution duration	30 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Resize instance

Resizing an EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Resizing an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-15mazjj88xc69" --change-type-version
 "2.0" --title "Resize EC2 Instance" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ResizeInstance\",\"Region\":\"ap-southeast-2\",\"Parameters
\":{\"InstanceId\":[\"i-0db3254017174df45\"],\"InstanceType\":[\"t2.xlarge\"],
\"CreateAMIBeforeResize\":[true]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it ResizeEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-15mazjj88xc69" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > ResizeEC2Params.json
   ```

2. Modify and save the ResizeEC2Params file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-ChangeInstanceType",
  "Region": "ap-southeast-2",
  "Parameters": {
    "InstanceId": [
      "i-0db3254017174df45"
    ],
    "InstanceType": [
      "t2.xlarge"
    ],
    "CreateAMIBeforeResize": [
      true
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ResizeEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ResizeEC2Rfc.json
```

4. Modify and save the ResizeEC2Rfc.json file. For example, you can replace the contents with something like this:.

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-15mazjj88xc69",
  "Title": "Resize EC2 Instance"
}
```

5. Create the RFC, specifying the ResizeEC2Rfc file and the ResizeEC2Params file:

```
aws amscm create-rfc --cli-input-json file://ResizeEC2Rfc.json  --execution-
parameters file://ResizeEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> Changing instance size can result in CloudFormation drift for any stacks that reference the changed instances.

For more information about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-15mazjj88xc69.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-ChangeInstanceType",
    "Region": "us-east-1",
    "Parameters": {
      "InstanceId": ["i-1234567890abababa"],
      "InstanceType": ["t3.xlarge"],
      "CreateAMIBeforeResize": [false]
    }
}
```

EC2 Instance Stack | Restore Volumes

Replace the instance volumes from an existing backup image of the instance.

Full classification: Management | Advanced stack components | EC2 instance stack | Restore volumes

Change Type Details

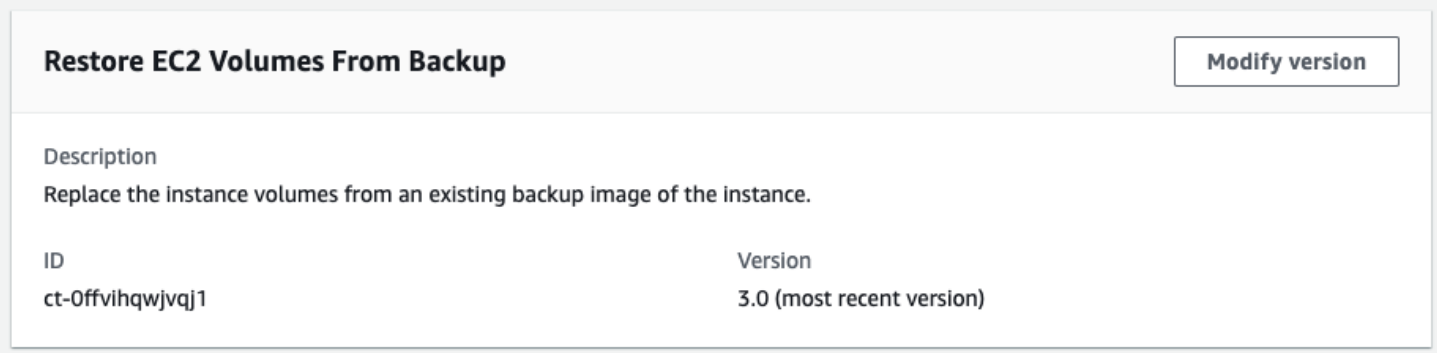
Change type ID	ct-0ffvihqwjvqj1
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Restore stack volumes

Restoring EC2 instance volumes with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Restoring EC2 instance volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0ffvihqwjvqj1" --change-type-version "3.0"
 --title "Restore EC2 Volumes From Backup" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ReplaceInstanceVolumesFromSnapshotsWithContext\",\"Region\":\"us-
east-1\",\"Parameters\":{\"InstanceId\":[\"INSTANCE_ID\"],\"Backup\":[\"BACKUP\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it RestoreEC2VolsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ffvihqwjvqj1" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RestoreEC2VolsParams.json
   ```

2. Modify and save the RestoreEC2VolsParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-
   ReplaceInstanceVolumesFromSnapshotsWithContext",
     "Region": "us-east-1",
     "Parameters": {
       "InstanceId": [
         "INSTANCE_ID"
       ],
   ```

```
    "Backup": [
      "EC2_BACKUP_ARN"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it RestoreEC2VolsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RestoreEC2VolsRfc.json
```

4. Modify and save the RestoreEC2VolsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-0ffvihqwjvqj1",
"Title":                "EC2-Restore-Volume-RFC"
}
```

5. Create the RFC, specifying the RestoreEC2VolsRfc file and the RestoreEC2VolsParams file:

```
aws amscm create-rfc --cli-input-json file://RestoreEC2VolsRfc.json  --execution-
parameters file://RestoreEC2VolsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If the snapshot ID of the instance volumes matches the snapshot ID associated with the provided AMI, then the volumes aren't restored.

For more information about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

For information on the automatic troubleshooting RFC created if this RFC fails, see EC2 instance volume restore fail

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ffvihqwjvqj1.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-
ReplaceInstanceVolumesFromSnapshotsWithContext",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceId": [
            "i-12345678"
        ],
        "Backup": [
            "ami-0ecdf967356c809c7"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-
ReplaceInstanceVolumesFromSnapshotsWithContext",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceId": [
            "i-12345678"
        ],
        "Backup": [
            "ami-0ecdf967356c809c7"
        ],
        "KMSKeyId": [
            "arn:aws:kms:us-east-1:123456789012:key/6f0a9efd-e1b7-41a2-
b04c-75fd89d9dc17"
        ],
        "ChangeHostname" : ["False"],
        "SleepTime": [
```

```
            "PT5M"
        ]
    }
}
```

EC2 Instance Stack | Start

Start up to 50 stopped EC2 instances.

Full classification: Management | Advanced stack components | EC2 instance stack | Start

Change Type Details

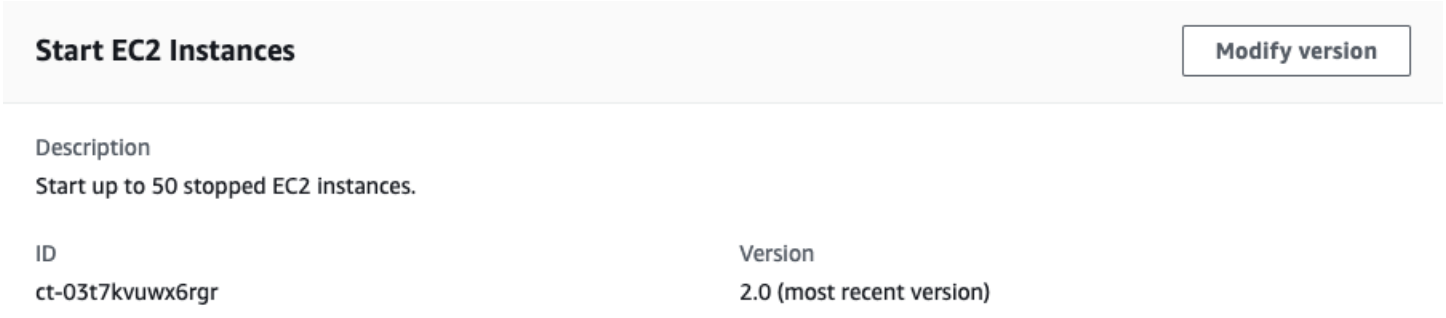
Change type ID	ct-03t7kvuwx6rgr
Current version	2.0
Expected execution duration	150 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start stack

Starting EC2 instances with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-03t7kvuwx6rgr" --change-type-version
 "2.0" --title "Start EC2 Instances" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-StartInstances\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstanceIds\":[\"i-1234567890abcdef0\", \"i-1234567890abcdef1\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it StartEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-03t7kvuwx6rgr" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartEC2Params.json
   ```

2. Modify and save the StartEC2Params file.

```
{
"DocumentName" : "AWSManagedServices-StartInstances",
"Region" : "us-east-1",
"Parameters" : {
    "InstanceIds" : [
    "i-1234567890abcdef0",
    "i-1234567890abcdef1"
    ]
}
}
```

3. Output the RFC template to a file in your current folder; this example names it StartEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StartEC2Rfc.json
```

4. Modify and save the StartEC2Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "2.0",
"ChangeTypeId": "ct-03t7kvuwx6rgr",
"Title": "Start EC2 Instances"
}
```

5. Create the RFC, specifying the StartEC2Rfc file and the StartEC2Params file:

```
aws amscm create-rfc --cli-input-json file://StartEC2Rfc.json --execution-
parameters file://StartEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 2.0. The schema has been changed so you can start up to fifty EC2 instances.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-03t7kvuwx6rgr.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-StartInstances",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceIds": [
            "i-1234567890abcdef0"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StartInstances",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceIds": ["i-1234567890abcdef0"]
    }
}
```

EC2 Instance Stack | Stop

Stop up to 50 running EC2 instances. If you specify an EC2 instance that is part of an Auto Scaling group (ASG), the instance is terminated and replaced by the ASG. If not part of an ASG, the instance remains stopped, in the account, until started or deleted.

Full classification: Management | Advanced stack components | EC2 instance stack | Stop

Change Type Details

Change type ID	ct-3mvvt2zkyveqj
Current version	3.0
Expected execution duration	150 minutes

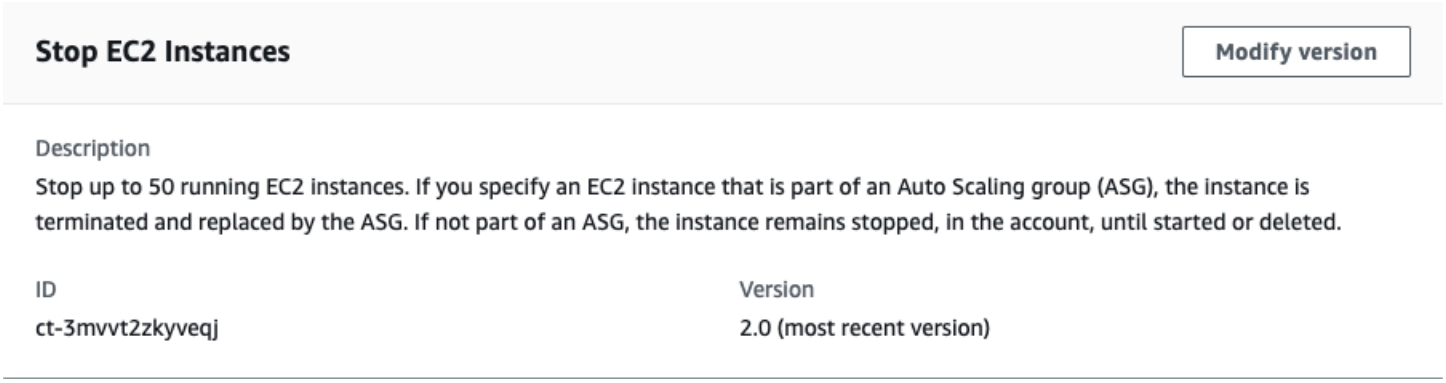
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop stack

Stopping an EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3mvvt2zkyveqj" --change-type-version
 "3.0" --title "Stop EC2 Instances" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-StopInstances\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstanceIds\":[\"i-1234567890abcdef0\", \"i-1234567890abcdef1\"],\"ForceStop\":
[\"false\"],\"StopASGInServiceInstances\":[\"false\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it StopEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3mvvt2zkyveqj" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopEC2Params.json
   ```

2. Modify and save the StopEC2Params file.

   ```
   {
   "DocumentName" : "AWSManagedServices-StopInstances",
   "Region" : "us-east-1",
   "Parameters" : {
       "InstanceIds" : [
       "i-1234567890abcdef0",
       "i-1234567890abcdef1"
       ],
       "ForceStop": [
       "false"
       ],
       "StopASGInServiceInstances": [
       "false"
       ]
   }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it StopEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StopEC2Rfc.json
```

4. Modify and save the StopEC2Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "3.0",
"ChangeTypeId": "ct-3mvvt2zkyveqj",
"Title": "Stop EC2 Instances"
}
```

5. Create the RFC, specifying the StopEC2Rfc file and the StopEC2Params file:

```
aws amscm create-rfc --cli-input-json file://StopEC2Rfc.json --execution-parameters
 file://StopEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 3.0. The schema has been changed so you can stop up to fifty instances.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3mvvt2zkyveqj.

Example: Required Parameters

```
{
```

```
    "DocumentName": "AWSManagedServices-StopInstances",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceIds": [
            "i-1234567890abcdef0"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StopInstances",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceIds": [
            "i-1234567890abcdef0"
        ],
        "ForceStop": [
            "false"
        ],
        "StopASGInServiceInstances": [
            "false"
        ]
    }
}
```

EC2 Instance Stack | Update

Use to modify the properties of an EC2 instance created using CT id ct-14027q0sjyt1h, version 3.0.

Full classification: Management | Advanced stack components | EC2 instance stack | Update

Change Type Details

Change type ID	ct-38s4s4tm4ic4u
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required

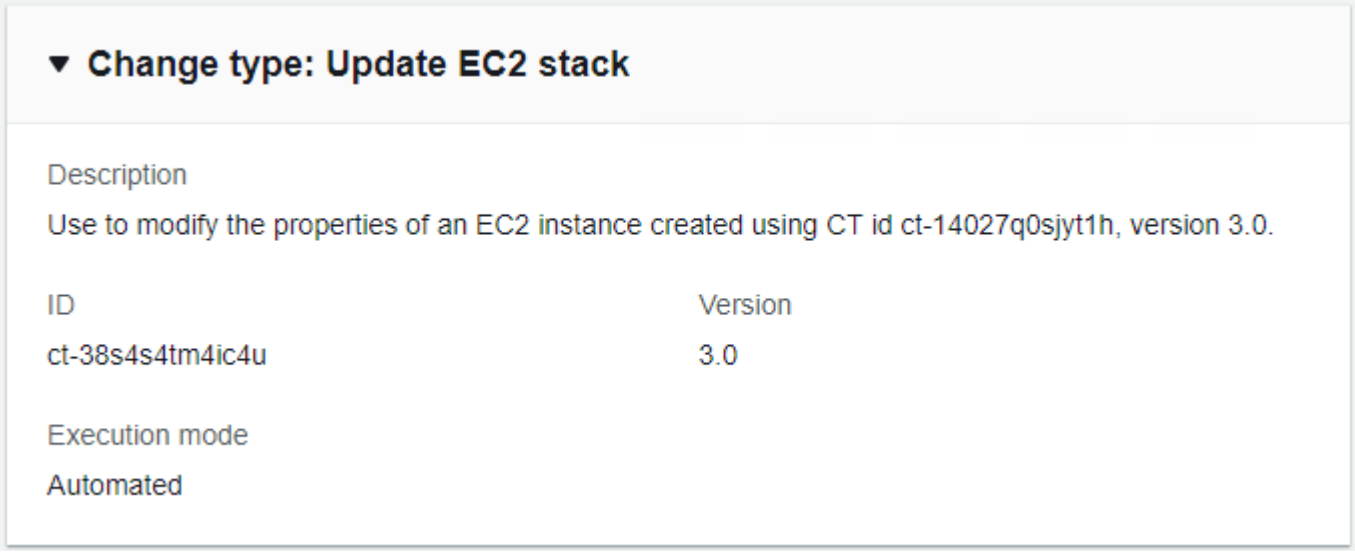
Customer approval	Not required
Execution mode	Automated

Additional Information

Update instances

Updating an EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Only specify the parameters you want to change. Absent parameters retain the existing values.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title test-ec2-stack-update --change-
type-id ct-38s4s4tm4ic4u --change-type-version 3.0 --execution-
parameters '{"VpcId":"VPC_ID","StackId":"STACK_ID","Parameters":
{"InstanceDetailedMonitoring":false,"InstanceEBSOptimized":false,"InstanceProfile":"customer-
mc-ec2-instance-profile","InstanceType":"t2.small","InstanceUserData":"#!/bin/bash\
\npwd\\nls -ltrh\\necho \"Hello, World\""}}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it UpdateEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-38s4s4tm4ic4u" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateEC2Params.json
   ```

2. Modify and save the UpdateEC2Params file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "VpcId": "VPC_ID",
     "StackId": "STACK_ID",
     "Parameters": {
       "InstanceDetailedMonitoring": false,
       "InstanceEBSOptimized": false,
       "InstanceProfile": "customer-mc-ec2-instance-profile",
       "InstanceType": "t2.small",
       "InstanceUserData": "#!/bin/bash\\npwd\\nls -ltrh\\necho \"Hello, World\""
   ```

```
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateEC2Rfc.json
```

4. Modify and save the UpdateEC2Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-38s4s4tm4ic4u",
"Title":                "EC2-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateEC2Rfc file and the UpdateEC2Params file:

```
aws amscm create-rfc --cli-input-json file://UpdateEC2Rfc.json  --execution-
parameters file://UpdateEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a version 3.0 change type and can be used to update EC2 instances created with the corresponding version 3.0 create change type, ct-14027q0sjyt1h.

To learn more about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-38s4s4tm4ic4u.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
  }
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
    "InstanceDetailedMonitoring": false,
    "InstanceEBSOptimized": false,
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceType": "t2.small",
    "InstanceUserData": "#!/bin/bash\\npwd\\nls -ltrh\\necho \"Hello, World\""
  }
}
```

EC2 Instance Stack | Update (With Additional Volumes)

Use to modify the properties of an EC2 instance created using CT id ct-1aqsjf86w6vxg, version 3.0.

Full classification: Management | Advanced stack components | EC2 instance stack | Update (with additional volumes)

Change Type Details

Change type ID	ct-1o1x2itfd6rk8
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required

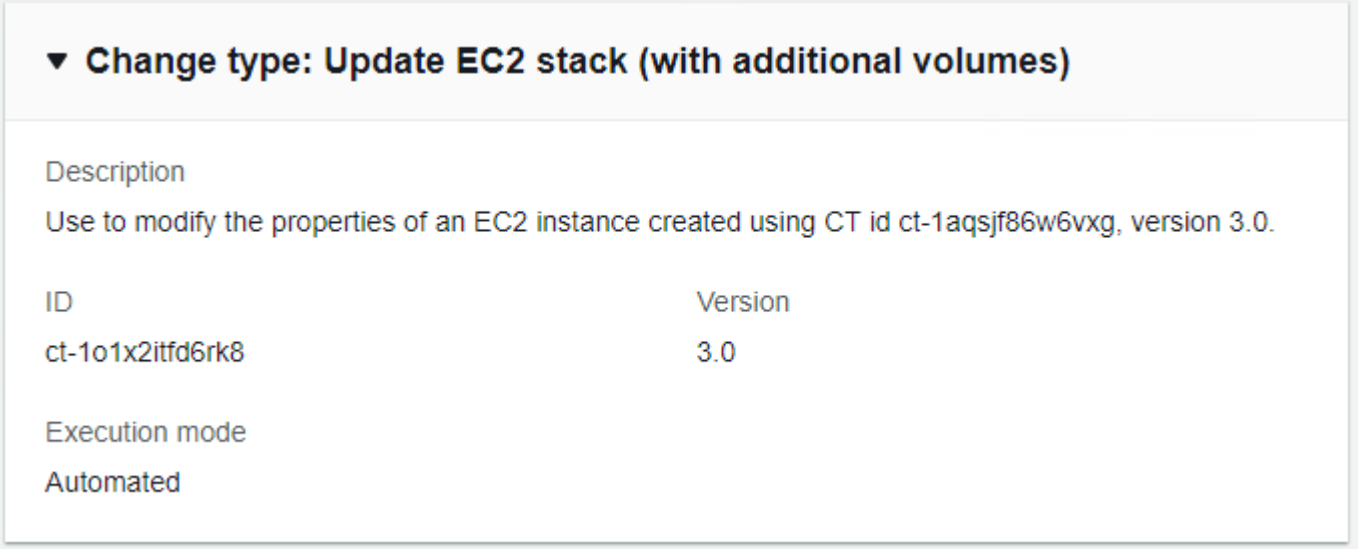
Execution mode	Automated

Additional Information

Update stack (with additional volumes)

Updating an EC2 Instance and Additional Volumes with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an EC2 Instance and Additional Volumes with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title test-ec2-stack-with-additional-volumes-
update --change-type-id ct-1o1x2itfd6rk8 --change-type-version 3.0 --
execution-parameters '{"VpcId":"VPC_ID","StackId":"STACK_ID","Parameters":
{"InstanceDetailedMonitoring":false,"InstanceEBSOptimized":false,"InstanceProfile":"customer-
mc-ec2-instance-profile","InstanceType":"t2.small","InstanceUserData":"#!/bin/bash\
\npwd\\nls -ltrh\\necho \"Hello,
 World\"","InstanceSecondaryPrivateIpAddressCount":1,"InstanceTerminationProtection":true,"Volu
dev/sdf","Volume1Size":100,"Volume1Type":"io1"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it UpdateEC2AVParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1o1x2itfd6rk8" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateEC2AVParams.json
   ```

2. Modify and save the UpdateEC2AVParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "EC2-Update-1-Addl-Volumes",
   "VpcId":            "VPC_ID",
   "Name":             "My-EC2-1-Addl-Volume",
   "TimeoutInMinutes": 60,
   "Parameters":   {
       "InstanceAmiId":    "AMI_ID",
       "InstanceSubnetId": "SUBNET_ID",
       "Volume1Encrypted": "true",
   ```

```
    "Volume1Iops":      "IOPS"
    "Volume1KmsKeyId":  "KMS_MASTER_KEY_ID",
    "Volume1Name":      "xvdh"
    "Volume1Size":      "2 GiB",
    "Volume1Snapshot":  "SNAPSHOT_ID",
    "Volume1Type":      "iol"
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateEC2AVRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateEC2AVRfc.json
```

4. Modify and save the UpdateEC2AVRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-1o1x2itfd6rk8",
"Title":                "EC2-Update-1-Addl-Volume-RFC"
}
```

5. Create the RFC, specifying the UpdateEC2AVRfc file and the UpdateEC2AVParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateEC2AVRfc.json  --execution-
parameters file://UpdateEC2AVParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a version 3.0 change type and can be used to update EC2 instances created with the corresponding version 3.0 create change type, ct-1aqsjf86w6vxg.

To learn more about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1o1x2itfd6rk8.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
  }
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
    "InstanceDetailedMonitoring": false,
    "InstanceEBSOptimized": false,
    "InstanceProfile": "customer-mc-ec2-instance-profile",
    "InstanceType": "t2.small",
    "InstanceUserData": "#!/bin/bash\\npwd\\nls -ltrh\\necho \"Hello, World\"",
    "InstanceSecondaryPrivateIpAddressCount": 1,
    "InstanceTerminationProtection": true,
    "Volume1Iops": 100,
    "Volume1Name": "/dev/sdf",
    "Volume1Size": 100,
    "Volume1Snapshot": "snap-1234567890abcdef0",
    "Volume1Type": "io1",
    "Volume2Iops": 100,
    "Volume2Name": "/dev/sdg",
    "Volume2Size": 100,
    "Volume2Snapshot": "snap-1234567890abcdef0",
    "Volume2Type": "io1",
    "Volume3Iops": 100,
    "Volume3Name": "/dev/sdh",
```

```
    "Volume3Size": 100,
    "Volume3Snapshot": "snap-1234567890abcdef0",
    "Volume3Type": "io1",
    "Volume4Iops": 100,
    "Volume4Name": "/dev/sdi",
    "Volume4Size": 100,
    "Volume4Snapshot": "snap-1234567890abcdef0",
    "Volume4Type": "io1",
    "Volume5Iops": 100,
    "Volume5Name": "/dev/sdj",
    "Volume5Size": 100,
    "Volume5Snapshot": "snap-1234567890abcdef0",
    "Volume5Type": "io1"
  }
}
```

EC2 Instance Stack | Update DeleteOnTermination (Review Required)

Update the EBS volume DeleteOnTermination property of the specified EC2 instance devices.

Full classification: Management | Advanced stack components | EC2 instance stack | Update DeleteOnTermination (review required)

Change Type Details

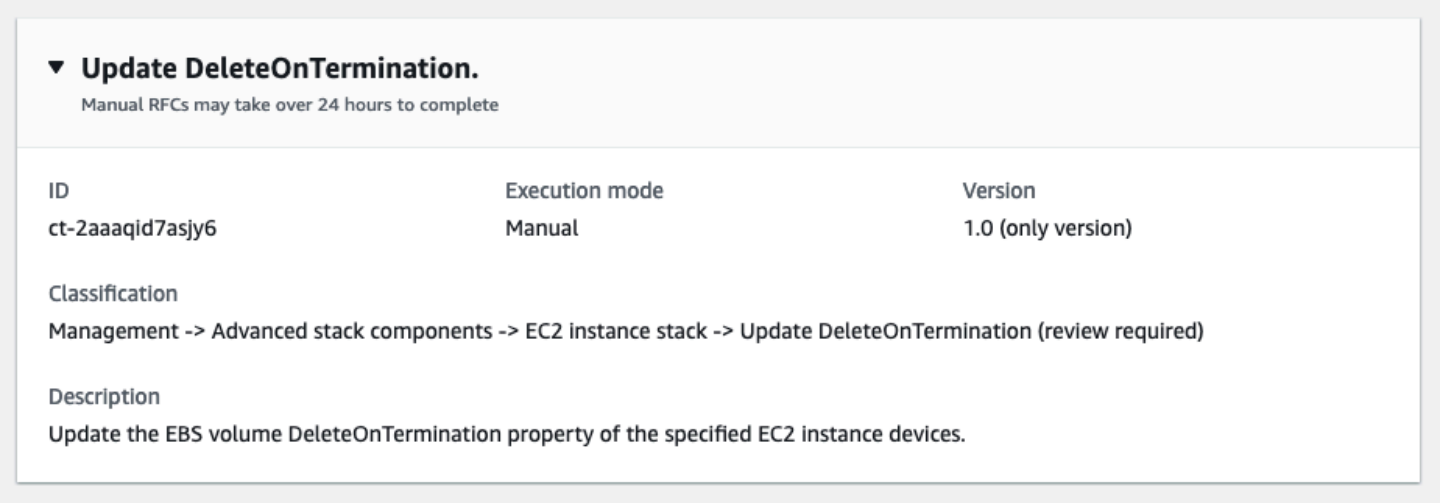
Change type ID	ct-2aaaqid7asjy6
Current version	1.0
Expected execution duration	30 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update the DeleteOnTermination option (review required)

Updating the DeleteOnTermination option with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating the DeleteOnTermination option with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2aaaqid7asjy6" --change-type-version
 "1.0" --title "Update DeleteOnTermination" --execution-parameters "{\"InstanceId
\": \"i-1234567890abcdef0\", \"DeviceNames\": [\"/dev/sda1\", \"/dev/xvda\"],
 \"DeleteOnTermination\": \"False\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it UpdateDeleteOnTerminationParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2aaaqid7asjy6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateDeleteOnTerminationParames.json
   ```

2. Modify and save the UpdateDeleteOnTerminationParams.json file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "InstanceId": "i-0cc489fa851c31a21",
     "DeviceNames": [
       "/dev/sda1",
       "/dev/xvda"
     ],
     "DeleteOnTermination": "False"
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdateDeleteOnTerminationRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateDeleteOnTerminationRfc.json
   ```

4. Modify and save the UpdateDeleteOnTerminationRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2aaaqid7asjy6",
"Title":                "Update DeleteOnTermination"
}
```

5. Create the RFC, specifying the UpdateDeleteOnTerminationRfc.json file and the UpdateDeleteOnTerminationParams.json file:

```
aws amscm create-rfc --cli-input-json file://UpdateDeleteOnTerminationRfc.json  --
execution-parameters file://UpdateDeleteOnTerminationParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EC2, including size recommendations, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2aaaqid7asjy6.

Example: Required Parameters

```
{
  "InstanceId": "i-1234567890abcdef0",
  "DeviceNames": ["/dev/sda", "/dev/xvda"],
  "DeleteOnTermination": "True"
}
```

Example: All Parameters

```
{
```

```
  "InstanceId": "i-1234567890abcdef0",
  "DeviceNames": ["/dev/sda", "/dev/xvda"],
  "DeleteOnTermination": "True"
}
```

EC2 Instance Stack | Update Instance Detailed Monitoring

Update EC2 instances' detailed monitoring setting through direct API calls. The EC2 instances can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-38s4s4tm4ic4u instead, or ct-361tlo1k7339x if the EC2 instance was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | EC2 instance stack | Update instance detailed monitoring

Change Type Details

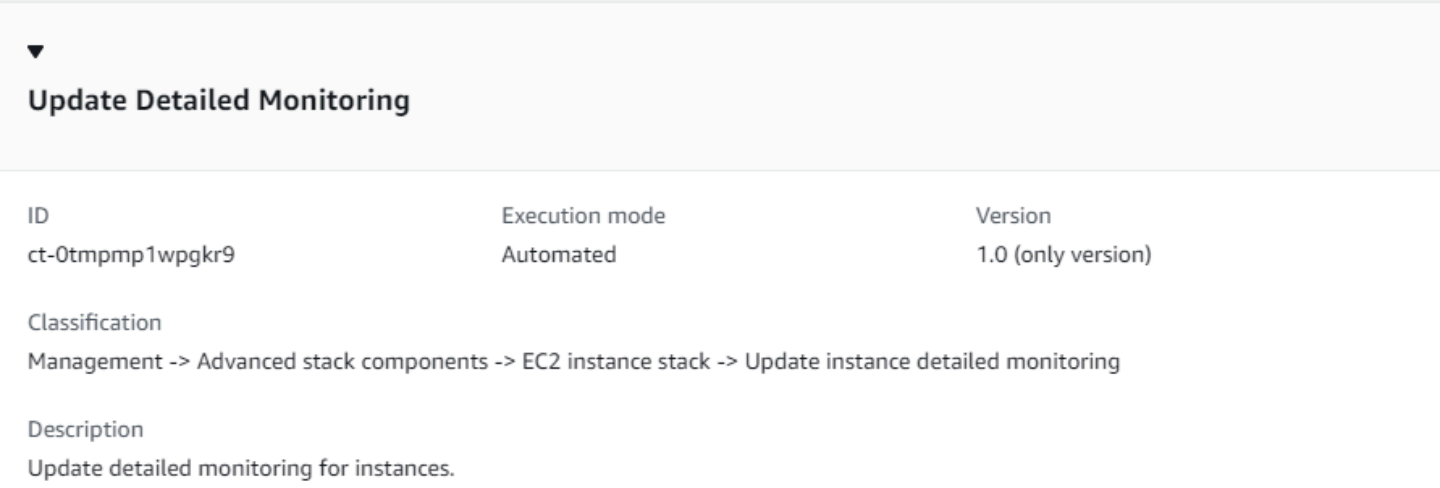
Change type ID	ct-0tmpmp1wpgkr9
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update detailed monitoring

Updating EC2 instances with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title "Update EC2 detailed monitoring" -update --change-
type-id ct-0tmpmp1wpgkr9 --change-type-version 1.0 --execution-parameters
 '{"DocumentName":"AWSManagedServices-UpdateInstanceEnhancedMonitoring","Region":"us-
east-1","Parameters":{"InstanceIds":["i-09d65b13db992e8d4","i-0cdbd78ad80d2378c"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it UpdateEc2MonitoringParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0tmpmp1wpgkr9"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateEc2MonitoringParams.json
   ```

2. Modify and save the UpdateEc2MonitoringParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateInstanceEnhancedMonitoring",
     "Region": "us-east-1",
     "Parameters": {
       "InstanceIds": [
         "i-09d65b13db992e8d4",
         "i-0cdbd78ad80d2378c"
       ],
       "MonitoringValue": "enabled"
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdateEc2MonitoringRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateEc2MonitoringRfc.json
   ```

4. Modify and save the UpdateEc2MonitoringRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0tmpmp1wpgkr9",
   "Title":                "EC2 Update Detailed Monitoring"
   ```

```
}
```

5. Create the RFC, specifying the UpdateEc2MonitoringRfc file and the UpdateEc2MonitoringParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateEc2MonitoringRfc.json  --
execution-parameters file://UpdateEc2MonitoringParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon EC2, see Amazon Elastic Compute Cloud Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0tmpmp1wpgkr9.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateInstanceEnhancedMonitoring",
  "Region": "eu-west-1",
  "Parameters": {
    "InstanceIds": ["i-1234567890abcdef0", "i-b188560f"],
    "MonitoringValue": ["enabled"]
  }
}
```

EC2 Instance Stack | Update Termination Protection

Update existing defined termination protection for EC2 instances.

Full classification: Management | Advanced stack components | EC2 instance stack | Update termination protection

Change Type Details

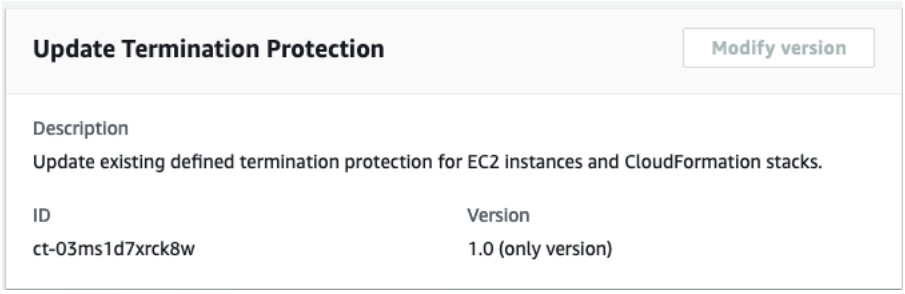
Change type ID	ct-03ms1d7xrck8w
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update stack termination protection

Updating an EC2 termination protection instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an EC2 instance termination protection with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Only specify the parameters you want to change. Absent parameters retain the existing values.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-03ms1d7xrck8w" \
--change-type-version "1.0" \
--title "Enable termination protection on EC2 instance" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ManageResourceTerminationProtection\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ResourceId\":[\"i-0d7e0c222654fc8f7\"],\"TerminationProtectionDesiredState\":
[\"enabled\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it UpdateTermProEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-38s4s4tm4ic4u"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateTermProEC2Params.json
   ```

2. Modify and save the UpdateTermProEC2Params file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
  "Region": "us-east-1",
  "Parameters": {
    "ResourceId": ["i-0d7e0c222654fc8f7"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateTermProEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateTermProEC2Rfc.json
```

4. Modify and save the UpdateTermProEC2Rfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-03ms1d7xrck8w",
    "ChangeTypeVersion": "1.0",
    "Title": "Enable termination protection on EC2 instance"
}
```

5. Create the RFC, specifying the UpdateTermProEC2Rfc file and the UpdateTermProEC2Params file:

```
aws amscm create-rfc --cli-input-json file://UpdateTermProEC2Rfc.json  --execution-
parameters file://UpdateTermProEC2Params.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> There is a related CT for AWS CloudFormation stacks, RDS Database Stack | Create.

To learn more about termination protection, see How do I protect my data against accidental EC2 instance termination?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-03ms1d7xrck8w.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
  "Region": "eu-west-1",
  "Parameters": {
    "ResourceId": ["i-1234567890"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

Identity and Access Management (IAM) | Delete Account Alias

Delete an existing AWS account alias. Note that if you delete the account alias, any URL containing the account alias stops working.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Delete account alias

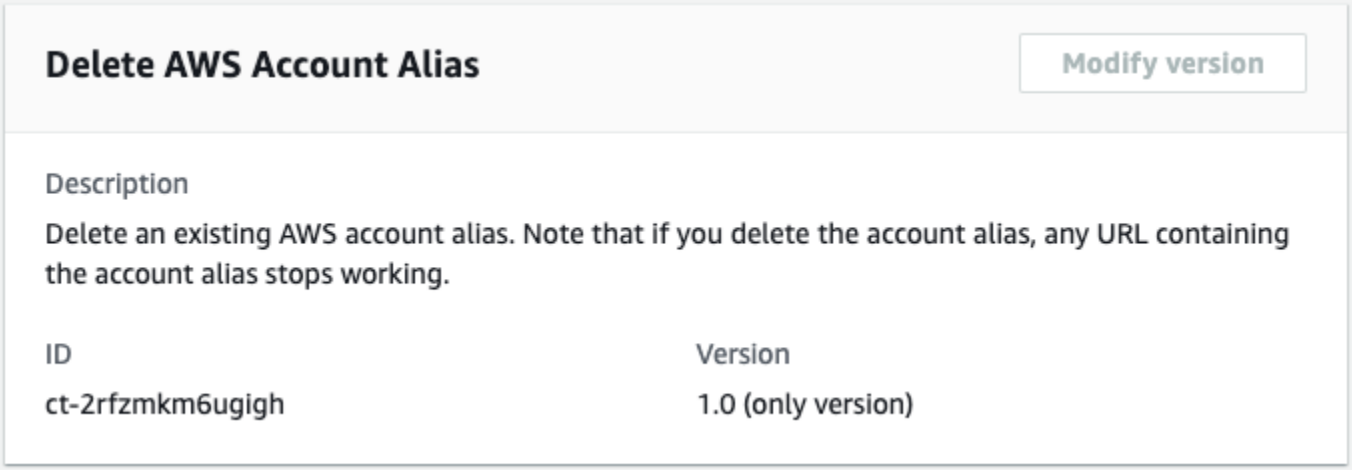
Change Type Details

Change type ID	ct-2rfzmkm6ugigh
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete IAM account alias

Deleting IAM account alias with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting IAM account alias with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2rfzmkm6ugigh" --change-type-
version "1.0" --title "Delete Account Alias" --execution-parameters
 '{"DocumentName":"AWSManagedServices-DeleteAccountAlias","Region":"us-
east-1","Parameters":{"AWSAccountAlias":["my-alias"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it DeleteIamAccountAliasParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2rfzmkm6ugigh"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteIamAccountAliasParams.json
   ```

2. Modify and save the DeleteIamAccountAliasParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-DeleteAccountAlias",
       "Region": "us-east-1",
       "Parameters": {
           "AWSAccountAlias": [
               "my-alias"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it DeleteIamAccountAliasRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteIamAccountAliasRfc.json
   ```

4. Modify and save the DeleteIamAccountAliasRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-2rfzmkm6ugigh",
    "ChangeTypeVersion": "1.0",
    "Title": "Delete Account Alias"
}
```

5. Create the RFC, specifying the DeleteIamAccountAliasRfc file and the DeleteIamAccountAliasParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteIamAccountAliasRfc.json  --
execution-parameters file://DeleteIamAccountAliasParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about IAM, see the IAM User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2rfzmkm6ugigh.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteAccountAlias",
  "Region": "us-east-1",
  "Parameters": {
    "AWSAccountAlias": ["myalias"]
  }
}
```

Identity and Access Management (IAM) | Delete Entity or Policy (Read-Write Permissions)

Delete Identity and Access Management (IAM) role or policy created with change type ct-1n9gfnog5x7fl.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Delete entity or policy (read-write permissions)

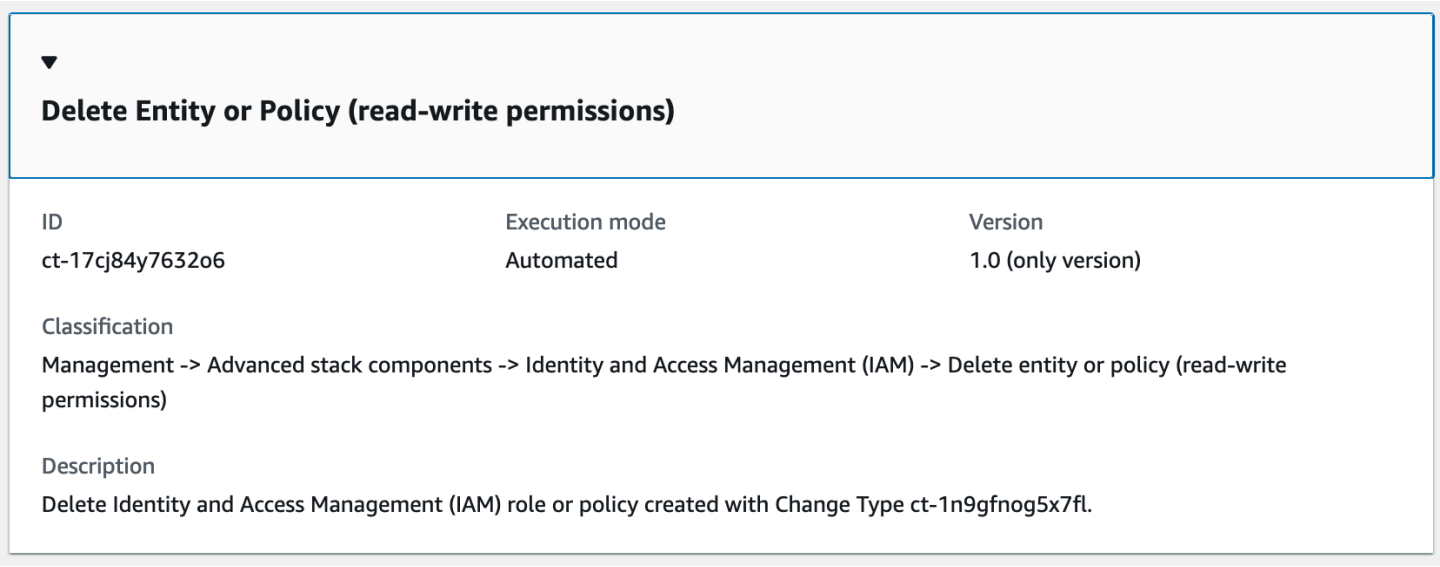
Change Type Details

Change type ID	ct-17cj84y7632o6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Automated

Additional Information

Delete IAM entity or policy

Deleting IAM entity or policy with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting IAM entity or policy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-17cj84y7632o6" --change-type-version
 "1.0" --title "Delete role or policy" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin\",\"Region
\":\"us-east-1\",\"Parameters\":{\"RoleName\":[\"TestRole01\",\"TestRole02\"],
\"ManagedPolicyName\":[\"TestPolicy01\",\"TestPolicy02\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it DeleteIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-17cj84y7632o6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteIamResourceParams.json
   ```

2. Modify and save the DeleteIamResourceParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin",
     "Region" : "us-east-1",
     "Parameters": {
       "RoleName": ["TestRole01", "TestRole02"],
       "ManagedPolicyName": ["TestPolicy01", "TestPolicy02"]
     }
   }
   ```

3. Output the RFC template JSON file to a file named DeleteIamResourceRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteIamResourceRfc.json
   ```

4. Modify and save the DeleteIamResourceRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
       "ChangeTypeVersion": "1.0",
   ```

```
    "ChangeTypeId": "ct-17cj84y7632o6",
    "Title": "Delete entity or policy (read-write permissions)"
}
```

5. Create the RFC, specifying the DeleteIamResourceRfc file and the DeleteIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteIamResourceRfc.json  --
execution-parameters file://DeleteIamResourceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-17cj84y7632o6.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin",
  "Region" : "us-east-1",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RoleName": ["TestRole01", "TestRole02"],
    "ManagedPolicyName": ["TestPolicy01", "TestPolicy02"]
  }
```

```
}
```

Identity and Access Management (IAM) | Delete Entity or Policy (Review Required)

Delete Identity and Access Management (IAM) users, roles or policies.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Delete entity or policy (review required)

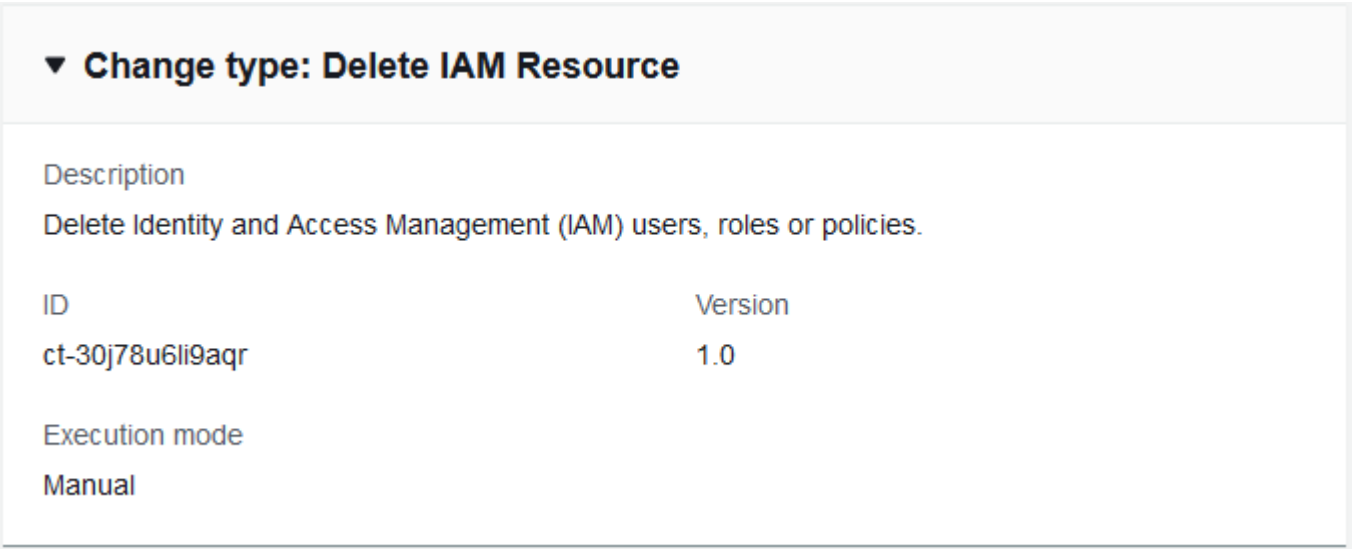
Change Type Details

Change type ID	ct-30j78u6li9aqr
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete IAM entity or policy

Deleting IAM Resources with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting IAM Resources with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-30j78u6li9aqr" --change-type-
version "1.0" --title "TestIamDelete" --execution-parameters "{\"IAM Roles\":
[\"arn:aws:iam::012345678901:role/test_role1\",\"arn:aws:iam::012345678901:role/
test_role2\"],\"Operation\":\"Delete\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it DeleteIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-30j78u6li9aqr"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteIamResourceParams.json
   ```

2. Modify and save the DeleteIamResourceParams file. For example, you can replace the contents with something like this:

```
{
  "IAM Roles": [
    "arn:aws:iam::012345678901:role/test_role1",
    "arn:aws:iam::012345678901:role/test_role2"
  ],
  "Operation": "Delete"
}
```

3. Output the RFC template JSON file to a file; this example names it DeleteIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteIamResourceRfc.json
```

4. Modify and save the DeleteIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-30j78u6li9aqr",
"Title": "Delete IAM roles"
}
```

5. Create the RFC, specifying the DeleteIamResourceRfc file and the DeleteIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteIamResourceRfc.json  --
execution-parameters file://DeleteIamResourceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> When using manual (approval required) CTs, AMS recommends that you use the ASAP option (choose ASAP in the console, leave start and end time blank in the API/CLI) as these

CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-30j78u6li9aqr.

Example: Required Parameters

```
{
  "Operation": "Delete"
}
```

Example: All Parameters

```
{
  "IAM Users": [
    "arn:aws:iam::012345678901:user/!\"#$%&'()*+,-.0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~/UsEr_+=,.@-"
  ],
  "IAM Roles": [
    "arn:aws:iam::012345678901:role/!\"#$%&'()*+,-.0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~/RoLe_+=,.@-"
  ],
  "IAM Policies": [
    "arn:aws:iam::012345678901:policy/!\"#$%&'()*+,-.0123456789:;<=>?
@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~/PoLiCy_+=,.@-"
  ],
  "Operation": "Delete",
  "Priority": "Medium"
}
```

Identity and Access Management (IAM) | Delete or Deactivate Access Key

Delete or deactivate the specified AWS IAM access key ID for the specified user.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Delete or deactivate access key

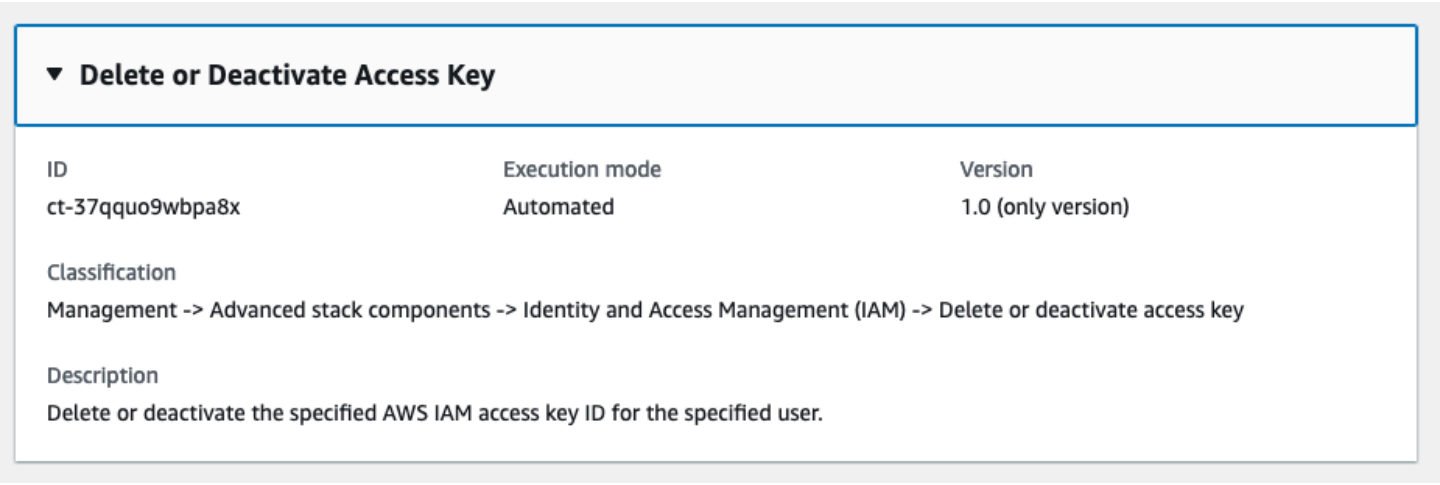
Change Type Details

Change type ID	ct-37qquo9wbpa8x
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete or deactivate access key

Deleting or deactivating access key with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting or deactivating access key with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Note

When pasting in a policy document, note that the RFC only accepts policy pastes up to 5,000 characters. If your file has more than 5,000 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-37qquo9wbpa8x" --change-type-version "1.0"
 --title "Delete or deactivate access key" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-DeactivateIAMAccessKey\",\"Region\": \"us-east-1\",\"Parameters
\": {\"UserName\": \"test-user\", \"AccessKeyId\": \"AKIAIOSFODNN7EXAMPLE\", \"Delete
\": false}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it DeactivateIamAccessKeyParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-37qquo9wbpa8x"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeactivateIamAccessKeyParams.json
```

2. Modify and save the DeactivateIamAccessKey file; example creates an IAM Role with policy documents pasted inline.

```
{
  "DocumentName": "AWSManagedServices-DeactivateIAMAccessKey",
  "Region": "us-east-1",
  "Parameters": {
    "UserName": "test-user",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Delete": false
  }
  }
}
```

3. Output the RFC template JSON file to a file named DeactivateIamAccessKeyRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeactivateIamAccessKeyRfc.json
```

4. Modify and save the DeactivateIamAccessKeyRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeVersion": "1.0",
    "ChangeTypeId": "ct-37qquo9wbpa8x",
    "Title": "Delete or Deactivate Access Key"
}
```

5. Create the RFC, specifying the DeactivateIamAccessKeyRfc.json file and the CreateIamResourceNrrParams file:

```
aws amscm create-rfc --cli-input-json file://DeactivateIamAccessKeyRfc.json --
execution-parameters file://DeactivateIamAccessKeyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-37qquo9wbpa8x.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DeactivateIAMAccessKey",
  "Region": "us-east-1",
  "Parameters": {
    "UserName": "myusername",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Delete": true
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeactivateIAMAccessKey",
  "Region": "us-east-1",
  "Parameters": {
    "UserName": "myusername",
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "Delete": false
  }
}
```

Identity and Access Management (IAM) | Delete SAML Identity Provider

Delete a SAML identity provider (IdP). The given IdP must not be referenced in any IAM role and must not be the only IdP in the account.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Delete SAML identity provider

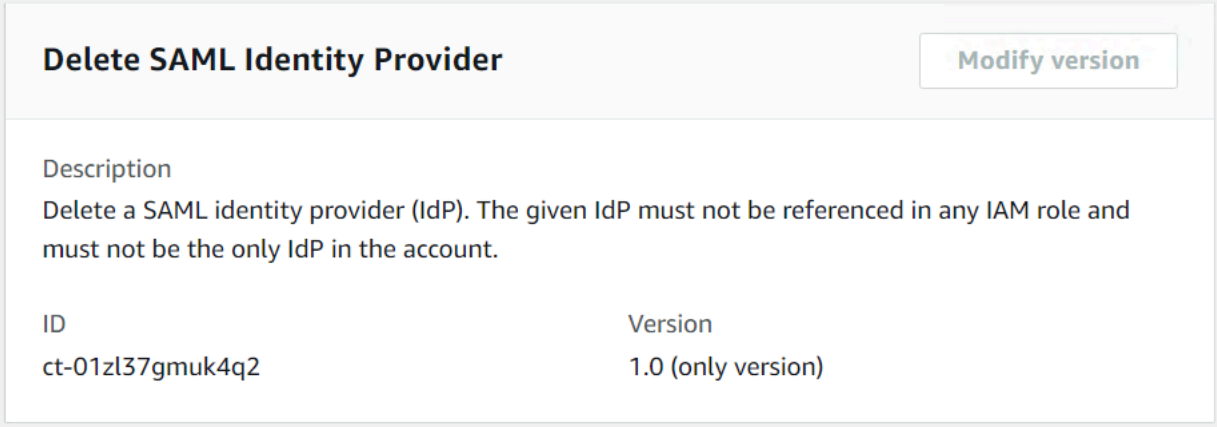
Change Type Details

Change type ID	ct-01zl37gmuk4q2
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete IAM SAML identity provider

Deleting IAM SAML IDPs with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting IAM SAML IDPs with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-01zl37gmuk4q2" --change-type-version "1.0"
 --title "Delete SAML Identity Provider" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleDeleteSamlProvider-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"Name\":[\"customer-saml\"],\"MetadataBackup\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it DeleteIamSamlIdpParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-01zl37gmuk4q2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteIamSamlIdpParams.json
   ```

2. Modify and save the DeleteIamSamlIdpParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName" : "AWSManagedServices-HandleDeleteSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Name" : [
      "customer-saml"
    ],
    "MetadataBackup": [
      "True"
    ]
  }
}
```

3. Output the RFC template JSON file to a file; this example names it DeleteIamSamlIdpRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteIamSamlIdpRfc.json
```

4. Modify and save the DeleteIamSamlIdpRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-01zl37gmuk4q2",
"Title": "Delete IAM SAML IDP"
}
```

5. Create the RFC, specifying the DeleteIamSamlIdpRfc file and the DeleteIamSamlIdpParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteIamSamlIdpRfc.json  --execution-
parameters file://DeleteIamSamlIdpParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about IAM, see the IAM User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-01zl37gmuk4q2.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleDeleteSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Name" : [
      "customer-saml"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleDeleteSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Name" : [
      "customer-saml"
    ],
    "MetadataBackup": [
      "True"
    ]
  }
}
```

Identity and Access Management (IAM) | Reset Service-Specific Credentials

Reset the password for the specified service-specific credential.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Reset service-specific credentials

Change Type Details

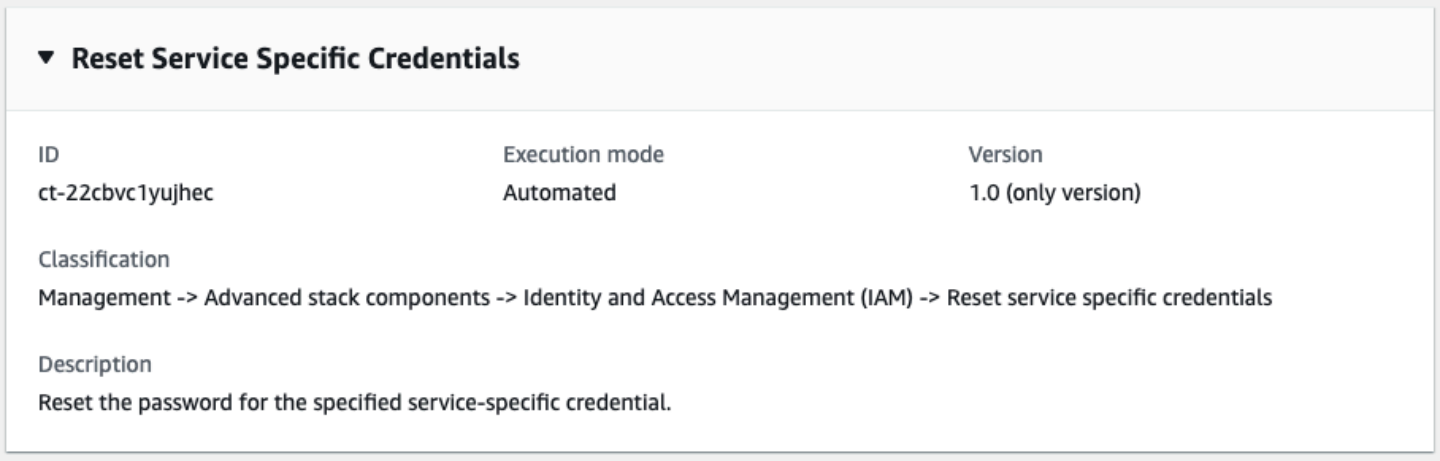
Change type ID	ct-22cbvc1yujhec

Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Reset service specific credentials

Resetting IAM service specific credentials with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Resetting IAM service specific credentials with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-22cbvc1yujhec" \
--change-type-version "1.0" --title "Reset service specific credentials for IAM User" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ResetServiceSpecificCredentials\",\"Region\":\"us-east-1\",\"Parameters\":{\"Username
\":[\"testuser\"],\"ServiceSpecificCredentialId\":[\"ACCAR712345678EXAMPLE\"],
\"SecretArn\":[\"arn:aws:secretsmanager:us-east-1:123456789012:secret:test-secret\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it ResetServSpecCredsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2ni31oyto1i5k"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ResetServSpecCredsParams.json
   ```

2. Modify and save the ResetServSpecCredsParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName" : "AWSManagedServices-ResetServiceSpecificCredentials",
     "Region" : "us-east-1",
     "Parameters" : {
   ```

```
    "Username" : [
      "testuser"
    ],
    "ServiceSpecificCredentialId" : [
      "ACCAR712345678EXAMPLE"
    ],
    "SecretArn" : [
      "arn:aws:secretsmanager:us-east-1:123456789012:secret:test-secret"
    ]
  }
}
```

3. Output the RFC template JSON file to a file named ResetServSpecCredsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ResetServSpecCredsRfc.json
```

4. Modify and save the ResetServSpecCredsRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-22cbvc1yujhec",
  "ChangeTypeVersion": "1.0",
  "Title": "Testing ct-22cbvc1yujhec ResetServiceSpecificCredentials in region us-
east-1 for an IAM User"
}
```

5. Create the RFC, specifying the ResetServSpecCredsRfc file and the ResetServSpecCredsParams file:

```
aws amscm create-rfc --cli-input-json file://ResetServSpecCredsRfc.json  --
execution-parameters file://ResetServSpecCredsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-22cbvc1yujhec.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-ResetServiceSpecificCredentials",
  "Region" : "us-east-1",
  "Parameters" : {
    "Username" : [
      "testuser"
    ],
    "ServiceSpecificCredentialId" : [
      "ACCAR712345678EXAMPLE"
    ],
    "SecretArn" : [
      "arn:aws:secretsmanager:us-east-1:123456789012:secret:test-secret"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-ResetServiceSpecificCredentials",
  "Region" : "us-east-1",
  "Parameters" : {
    "Username" : [
      "testuser"
    ],
    "ServiceSpecificCredentialId" : [
      "ACCAR712345678EXAMPLE"
    ],
    "SecretArn" : [
      "arn:aws:secretsmanager:us-east-1:123456789012:secret:test-secret"
    ]
  }
}
```

Identity and Access Management (IAM) | Update Account Alias

Update an existing AWS account alias. Note that an AWS account can have only one alias. If you update the account alias, the new alias overwrites the previous alias, and the URL containing the previous alias stops working.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Update account alias

Change Type Details

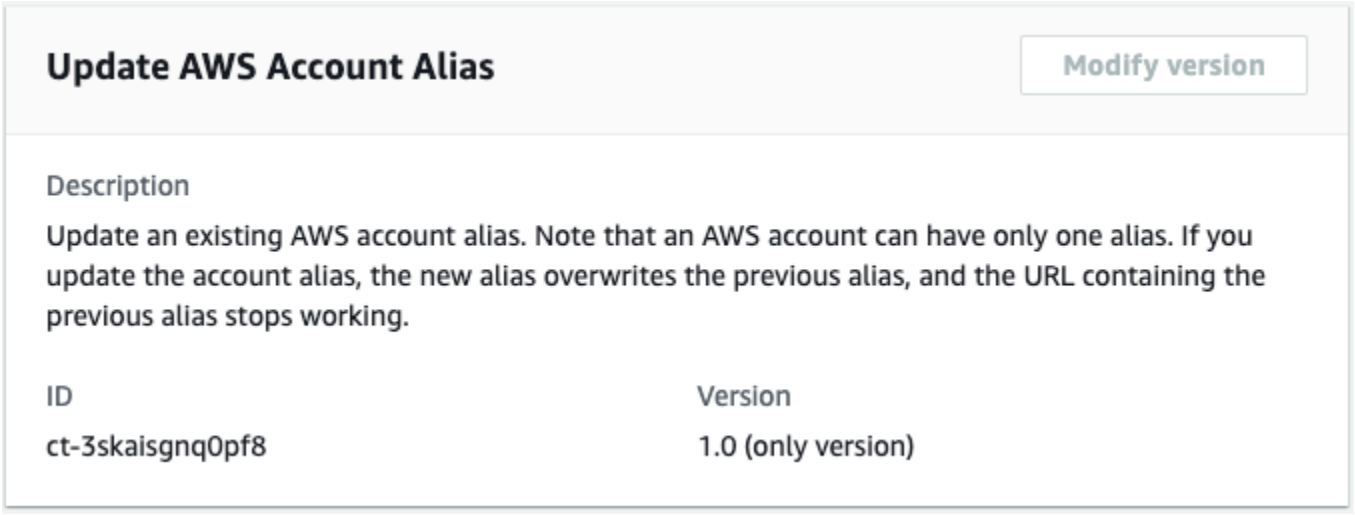
Change type ID	ct-3skaisgnq0pf8
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update IAM account alias

Updating IAM account alias with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating IAM account alias with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3skaisgnq0pf8" --change-type-
version "1.0" --title "Update Account Alias" --execution-parameters
 '{"DocumentName":"AWSManagedServices-CreateAccountAlias","Region":"us-
east-1","Parameters":{"AWSAccountAlias":["my-new-alias"], "ReplaceAliasIfExists":
["True"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateIamAccountAliasParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3skaisgnq0pf8"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateIamAccountAliasParams.json
   ```

2. Modify and save the UpdateIamAccountAliasParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-CreateAccountAlias",
       "Region": "us-east-1",
       "Parameters": {
           "AWSAccountAlias": [
               "my-new-alias"
           ],
           "ReplaceAliasIfExists": [
               "True"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it UpdateIamAccountAliasRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateIamAccountAliasRfc.json
   ```

4. Modify and save the UpdateIamAccountAliasRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
       "ChangeTypeId": "ct-3skaisgnq0pf8",
   ```

```
    "ChangeTypeVersion": "1.0",
    "Title": "Update Account Alias"
}
```

5. Create the RFC, specifying the UpdateIamAccountAliasRfc file and the UpdateIamAccountAliasParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateIamAccountAliasRfc.json  --
execution-parameters file://UpdateIamAccountAliasParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about IAM, see the IAM User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3skaisgnq0pf8.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateAccountAlias",
  "Region": "us-east-1",
  "Parameters": {
    "AWSAccountAlias": ["myalias"],
    "ReplaceAliasIfExists": ["True"]
  }
}
```

Identity and Access Management (IAM) | Update Entity or Policy (Read-Write Permissions)

Update Identity and Access Management (IAM) role or policy with read-write permissions. You must have enabled this feature with change type ct-1706xvvk6j9hf before submitting this request. Automated IAM provisioning with read-write permissions runs over 200 validations to help ensure successful outcomes.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Update entity or policy (read-write permissions)

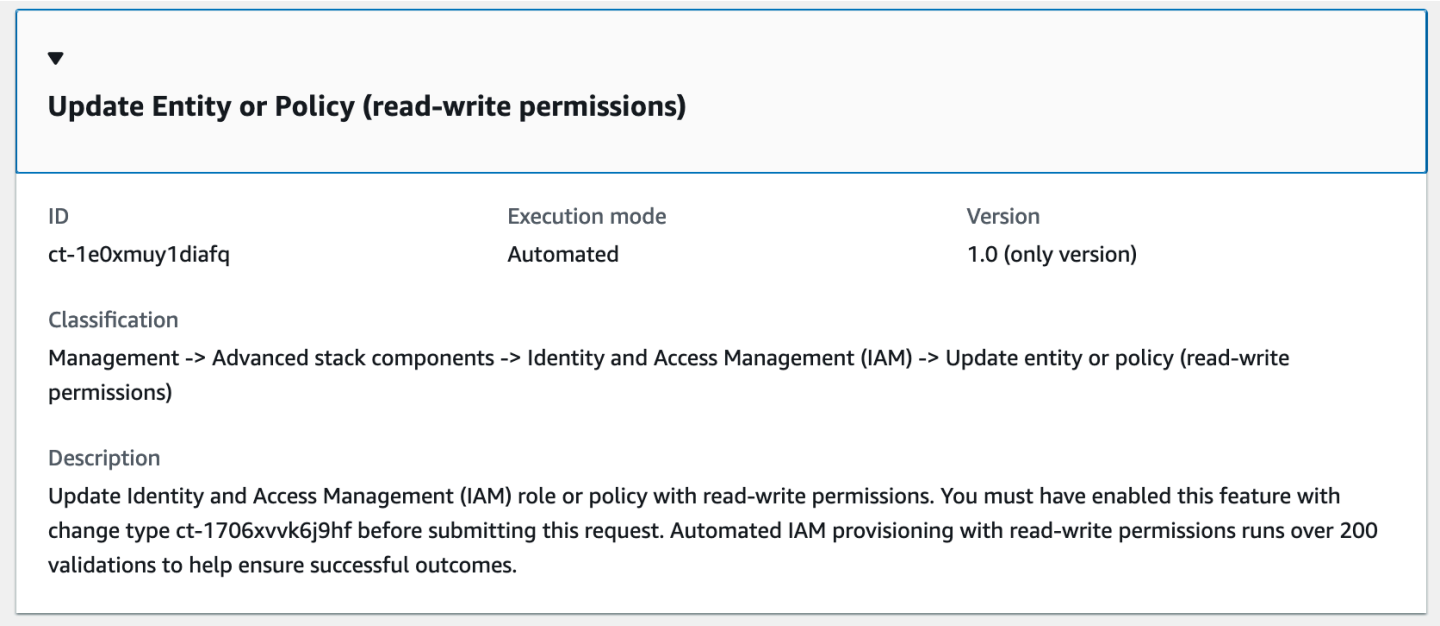
Change Type Details

Change type ID	ct-1e0xmuy1diafq
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Automated

Additional Information

Update IAM entity or policy

Updating IAM entity or policy with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating IAM entity or policy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1e0xmuy1diafq" --change-type-version
 "1.0" --title "Update role or policy" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin\",\"Region
\":\"us-east-1\",\"Parameters\":{\"ValidateOnly\":\"No\"},\"RoleDetails
\":{\"Roles\":[{\"RoleName\":\"RoleTest01\",\"Description\":\"This is a test
 role\",\"AssumeRolePolicyDocument\":\"{\\\"Version\\\":\\\"2012-10-17\\\",\
\\"Statement\\\":[{\\\"Effect\\\":\\\"Allow\\\",\\\"Principal\\\":{\\\"AWS\\
\":\\\"arn:aws:iam::123456789012:root\\\"},\\\"Action\\\":\\\"sts:AssumeRole\
\\"}]}\",\"ManagedPolicyArns\":[\"arn:aws:iam::123456789012:policy/policy01\",
\"arn:aws:iam::123456789012:policy/policy02\"],\"MaxSessionDuration\":\"7200\",
\"PermissionsBoundary\":\"arn:aws:iam::123456789012:policy/permission_boundary01\"}]},
\"ManagedPolicyDetails\":{\"Policies\":[{\"ManagedPolicyName\":\"TestPolicy01\",
\"PolicyDocument\":\"{\\\"Version\\\":\\\"2012-10-17\\\",\\\"Statement\\\":
[{\\\"Sid\\\":\\\"AllQueueActions\\\",\\\"Effect\\\":\\\"Allow\\\",\\\"Action
\\\":\\\"sqs:ListQueues\\\",\\\"Resource\\\":\\\"*\\\",\\\"Condition\\\":{\\
\"ForAllValues:StringEquals\\\":{\\\"aws:tagKeys\\\":[\\\"temporary\\\"]}}}]}\"}]}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it UpdateIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1e0xmuy1diafq"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateIamResourceParams.json
   ```

2. Modify and save the UpdateIamResourceParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin",
   ```

```
  "Region" : "us-east-1",
  "Parameters": {
    "ValidateOnly": "No"
  },
  "RoleDetails": {
    "Roles": [
      {
        "RoleName": "RoleTest01",
        "Description": "This is a test role",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}",
        "ManagedPolicyArns": [
          "arn:aws:iam::123456789012:policy/policy01",
          "arn:aws:iam::123456789012:policy/policy02"
        ],
        "MaxSessionDuration": "7200",
        "PermissionsBoundary": "arn:aws:iam::123456789012:policy/
permission_boundary01"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource
\":\"*\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary
\"]}}}]}"
      }
    ]
  }
}
```

3. Output the RFC template JSON file to a file named UpdateIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateIamResourceRfc.json
```

4. Modify and save the UpdateIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
```

```
    "ChangeTypeVersion": "1.0",
    "ChangeTypeId": "ct-1e0xmuy1diafq",
    "Title": "Update entity or policy (read-write permissions)"
}
```

5. Create the RFC, specifying the UpdateIamResourceRfc file and the UpdateIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateIamResourceRfc.json  --
execution-parameters file://UpdateIamResourceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1e0xmuy1diafq.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin",
  "Region" : "us-east-1",
  "Parameters": {"ValidateOnly":  "No"}
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "ValidateOnly": "No"
```

```
  },
  "RoleDetails": {
    "Roles": [
      {
        "RoleName": "RoleTest01",
        "Description": "This is a test role",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}",
        "ManagedPolicyArns": [
          "arn:aws:iam::123456789012:policy/policy01",
          "arn:aws:iam::123456789012:policy/policy02"
        ],
        "MaxSessionDuration": "7200",
        "PermissionsBoundary": "arn:aws:iam::123456789012:policy/permission_boundary01"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource\":\"*
\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary\"]}}}]}"
      }
    ]
  }
}
```

Identity and Access Management (IAM) | Update Entity or Policy (Review Required)

Update Identity and Access Management (IAM) user, role, or policy.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Update entity or policy (review required)

Change Type Details

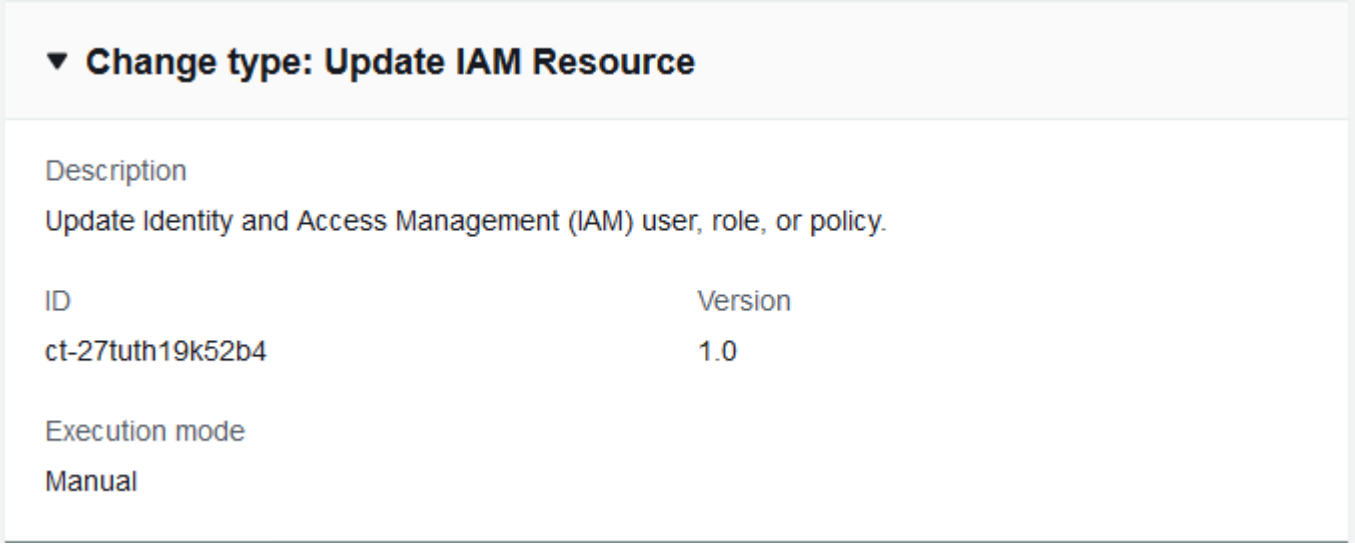
Change type ID	ct-27tuth19k52b4

Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update IAM entity or policy

Updating IAM Resources with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating IAM Resources with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-27tuth19k52b4" --change-type-version "1.0"
 --title "TestIamUpdate" --execution-parameters "{\"UseCase\":\"IAM_RESOURCE_DETAILS\",
\"IAM Role\":[{\"RoleName\":\"ROLE_NAME\",\"TrustPolicy\":\"TRUST_POLICY\",
\"RolePermissions\":\"ROLE_PERMISSIONS\"}],\"Operation\":\"Update\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateIamResourceParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-27tuth19k52b4"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateIamResourceParams.json
   ```

2. Modify and save the UpdateIamResourceParams file. For example, you can replace the contents with something like this:

   ```
   {
     "UseCase": "IAM_RESOURCE_DETAILS",
     "IAM Role": [
       {
         "RoleName": "codebuild_ec2_test_role",
   ```

```
      "TrustPolicy": "{\"Version\":\"2008-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Principal\":{\"Service\":\"codebuild.amazonaws.com\"},\"Action\":
\"sts:AssumeRole\"}]}",
      "RolePermissions": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Effect\":
\"Allow\",\"Action\":[\"ec2:DescribeInstanceStatus\"],\"Resource\":\"*\"}]}"
    }
  ],
  "Operation": "Update"
}
```

3. Output the RFC template JSON file to a file; this example names it UpdateIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateIamResourceRfc.json
```

4. Modify and save the UpdateIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-27tuth19k52b4",
"Title": "Update IAM Roles"
}
```

5. Create the RFC, specifying the UpdateIamResourceRfc file and the UpdateIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateIamResourceRfc.json  --
execution-parameters file://UpdateIamResourceParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- Important. We cannot update or modify AMS default and AMS self-service provisioning service (SSPS) IAM entities as per our technical standards, with some exceptions. As an alternative we can create a clone of these entities with a custom name and required permission set to deploy in your account.

- When using manual (approval required) CTs, AMS recommends that you use the ASAP option (choose ASAP in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run.
- We cannot update or modify AMS default or AMS self-service provisioning service (SSPS) IAM entities. If you require similar set of permissions that are provided in the default and SSPS IAM entities, we can create a clone of the entities with a custom name that you can provide us within the RFC execution parameters (**UseCase**).

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-27tuth19k52b4.

Example: Required Parameters

```
{
  "UseCase": "Use case...",
  "Operation": "Update"
}
```

Example: All Parameters

```
{
  "UseCase": "Use case...",
  "IAM User": [
    {
      "UserName": "user-a",
      "UserPermissions": "Power User permissions",
      "Tags": [
        {
          "Key": "foo",
          "Value": "bar"
        },
```

```
        {
          "Key": "testkey",
          "Value": "testvalue"
        }
      ]
    }
  ],
  "IAM Role": [
    {
      "RoleName": "role-b",
      "TrustPolicy": "Trust policy example",
      "RolePermissions": "Role permissions example",
      "Tags": [
        {
          "Key": "foo",
          "Value": "bar"
        },
        {
          "Key": "testkey",
          "Value": "testvalue"
        }
      ]
    }
  ],
  "IAM Policy": [
    {
      "PolicyName": "policy1",
      "PolicyDocument": "Policy document example 1",
      "RelatedResources": [
        "resourceA",
        "resourceB"
      ]
    },
    {
      "PolicyName": "policy2",
      "PolicyDocument": "Policy document example 2",
      "RelatedResources": [
        "resourceC",
        "resourceD"
      ]
    }
  ],
  "Operation": "Update",
  "Priority": "Medium"
```

```
}
```

Identity and Access Management (IAM) | Update MaxSessionDuration

Update the MaxSessionDuration property of an AWS Identity and Access Management (IAM) role. This setting determines the maximum duration that can be requested using the DurationSeconds parameter when assuming an IAM role.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Update MaxSessionDuration

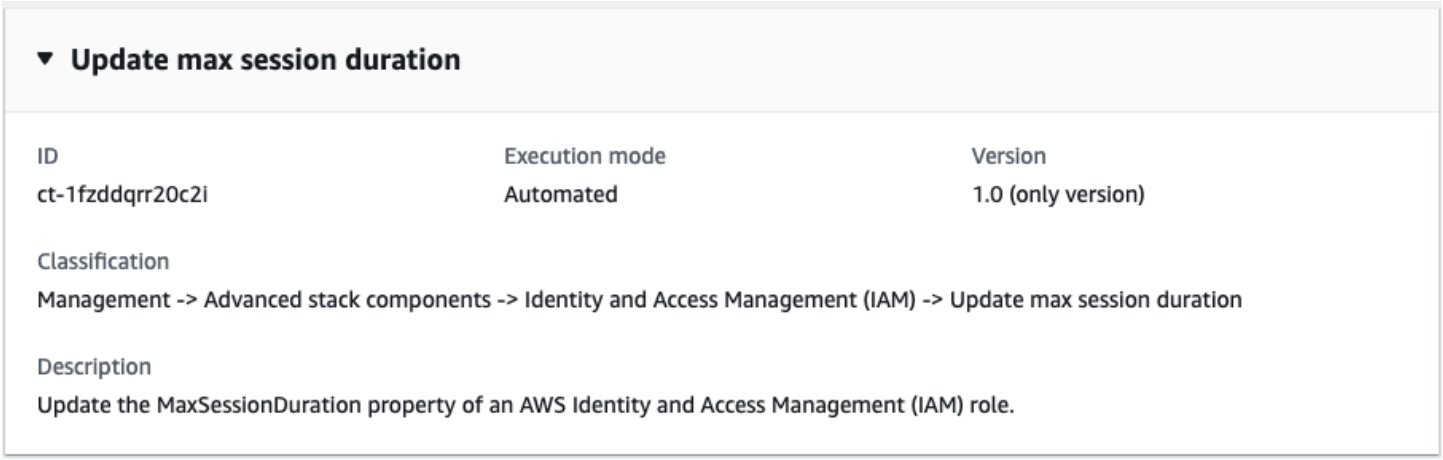
Change Type Details

Change type ID	ct-1fzddqrr20c2i
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update IAM role MaxSessionDuration

Updating an IAM Max Session Duration with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an IAM Max Session Duration with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

> **Note**
>
> When pasting in a policy document, note that the RFC only accepts policy pastes up to 5,000 characters. If your file has more than 5,000 characters, create a service request to upload the policy and then refer to that service request in the RFC that you open for IAM.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1fzddqrr20c2i" \
--change-type-version "1.0" --title "Update max session duration" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
UpdateIAMRoleMaxSessionDuration\",\"Region\":\"us-east-1\",\"Parameters\":{\"RoleName
\":[\"role-name\"],\"MaxSessionDuration\":[3600]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it UpdateMaxSessDurationParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1fzddqrr20c2i"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateMaxSessDurationParams.json
   ```

2. Modify and save the UpdateMaxSessDurationParams file; example creates an IAM Role with policy documents pasted inline.

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateIAMRoleMaxSessionDuration",
     "Region": "us-east-1",
     "Parameters": {
       "RoleName": [
         "role-name"
       ],
       "MaxSessionDuration": [
   ```

```
      3600
    ]
  }
}
```

3. Output the RFC template JSON file to a file named UpdateMaxSessDurationRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateMaxSessDurationRfc.json
```

4. Modify and save the UpdateMaxSessDurationRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1fzddqrr20c2i",
  "Title": "Update max session duration"
}
```

5. Create the RFC, specifying the UpdateMaxSessDurationRfc file and the UpdateMaxSessDurationParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateMaxSessDurationRfc.json  --
execution-parameters file://UpdateMaxSessDurationParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1fzddqrr20c2i.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Identity and Access Management (IAM) | Update SAML Identity Provider

Update IAM identity provider using the SAML metadata document file that you stored in your chosen S3 bucket.

Full classification: Management | Advanced stack components | Identity and Access Management (IAM) | Update SAML identity provider

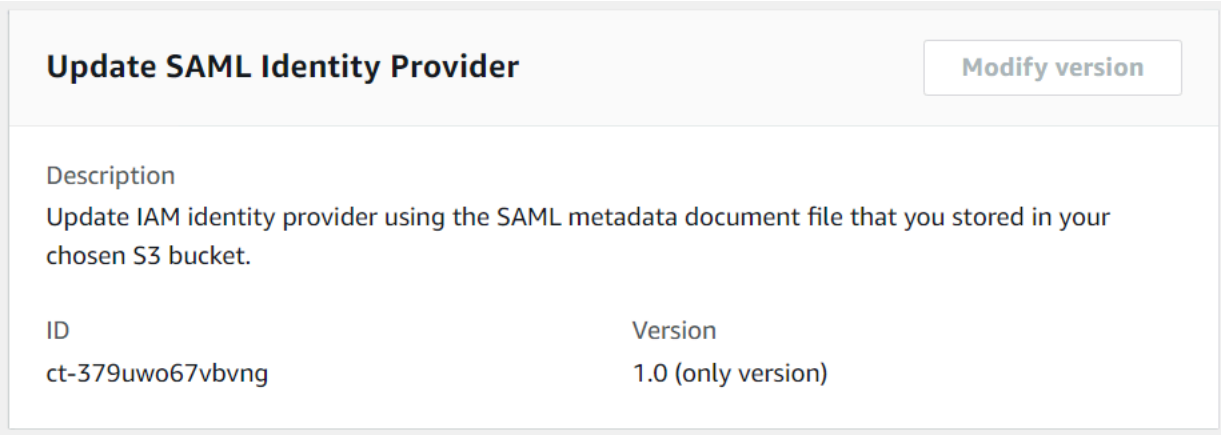
Change Type Details

Change type ID	ct-379uwo67vbvng
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update IAM SAML identity provider

Updating IAM SAML identity providers with the Console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating IAM SAML identity providers with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-379uwo67vbvng" --change-type-version "1.0"
 --title "Update SAML Identity Provider" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-HandleUpdateSamlProvider-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"SAMLMetadataDocumentURL\":[\"s3://bucket.name/idp-metadata.xml\"],
\"SAMLProviderArn\":[\"arn:aws:iam::123456789012:saml-provider/customer-saml\"],
\"SAMLProviderBackup\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateIamSamlIdpParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-379uwo67vbvng"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateIamSamlIdpParams.json
   ```

2. Modify and save the UpdateIamSamlIdpParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-HandleUpdateSamlProvider-Admin",
     "Region" : "us-east-1",
     "Parameters" : {
       "SAMLMetadataDocumentURL" : [
         "s3://bucket.name/idp-metadata.xml"
       ],
       "SAMLProviderArn" : [
         "arn:aws:iam::123456789012:saml-provider/customer-saml"
       ],
       "SAMLProviderBackup" : [
         "True"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it UpdateIamSamlIdpRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateIamSamlIdpRfc.json
```

4. Modify and save the UpdateIamSamlIdpRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-379uwo67vbvng",
"Title": "Update IAM SAML IDP"
}
```

5. Create the RFC, specifying the UpdateIamSamlIdpRfc file and the UpdateIamSamlIdpParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateIamSamlIdpRfc.json  --execution-
parameters file://UpdateIamSamlIdpParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-379uwo67vbvng.

Example: Required Parameters

```
{

  "DocumentName" : "AWSManagedServices-HandleUpdateSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SAMLMetadataDocumentURL" : [
      "s3://bucket/path/to/metadata.xml"
    ],
```

```
    "SAMLProviderArn" : [
      "arn:aws:iam::123456789012:saml-provider/customer-saml"
    ]
  }
}
```

Example: All Parameters

```
{

  "DocumentName" : "AWSManagedServices-HandleUpdateSamlProvider-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SAMLMetadataDocumentURL" : [
      "s3://bucket/path/to/metadata.xml"
    ],
    "SAMLProviderArn" : [
      "arn:aws:iam::123456789012:saml-provider/customer-saml"
    ],
    "SAMLProviderBackup" : [
      "True"
    ]
  }
}
```

KMS Alias | Delete

Delete an alias of an AWS Key Management Service (KMS) customer master key (CMK).

Full classification: Management | Advanced stack components | KMS alias | Delete

Change Type Details

Change type ID	ct-04gzyy008v1bg
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required

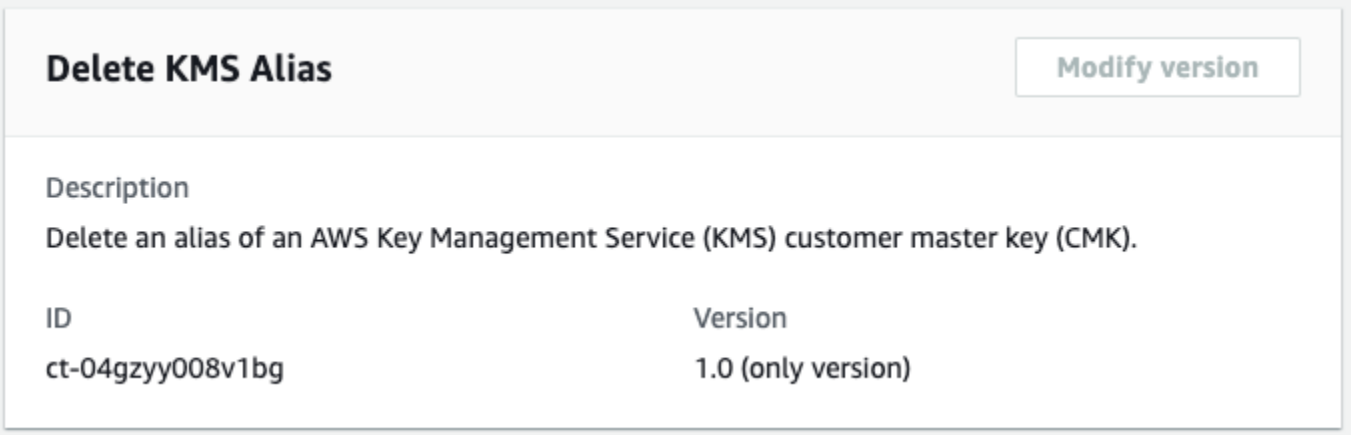
Execution mode	Automated

Additional Information

Delete an AWS KMS alias

Deleting an AWS KMS alias with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AWS KMS alias with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title delete-kms-alias --change-type-id ct-04gzyy008v1bg --
change-type-version 1.0 --execution-parameters '{"DocumentName": "AWSManagedServices-
DeleteKMSAlias", "Region": "us-east-1", "Parameters": {"AliasName": ["my-test-key"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteKmsAliasParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-04gzyy008v1bg" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteKmsAliasParams.json
   ```

2. Modify and save the DeleteKmsAliasParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DeleteKMSAlias",
     "Region": "us-east-1",
     "Parameters": {
       "AliasName": ["my-test-key"]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it DeleteKmsAliasRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteKmsAliasRfc.json
   ```

4. Modify and save the DeleteKmsAliasRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-04gzyy008v1bg",
"Title":                "delete-kms-alias"
}
```

5. Create the RFC, specifying the DeleteKmsAlias Rfc file and the DeleteKmsAliasParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteKmsAliasRfc.json  --execution-
parameters file://DeleteKmsAliasParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about KMS, see Key Management Service.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-04gzyy008v1bg.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteKMSAlias",
  "Region" : "us-east-1",
  "Parameters" : {
    "AliasName" : [
      "test-alias"
    ]
  }
}
```

Example: All Parameters

```
Example not available.
```

KMS Key | Delete (Review Required)

Delete an AWS Key Management Service (KMS) Key from an AMS account. By default, there is a 30 day waiting period before the key is deleted; during that period, you can restore the key using the KMS Key Update change type.

Full classification: Management | Advanced stack components | KMS key | Delete (review required)

Change Type Details

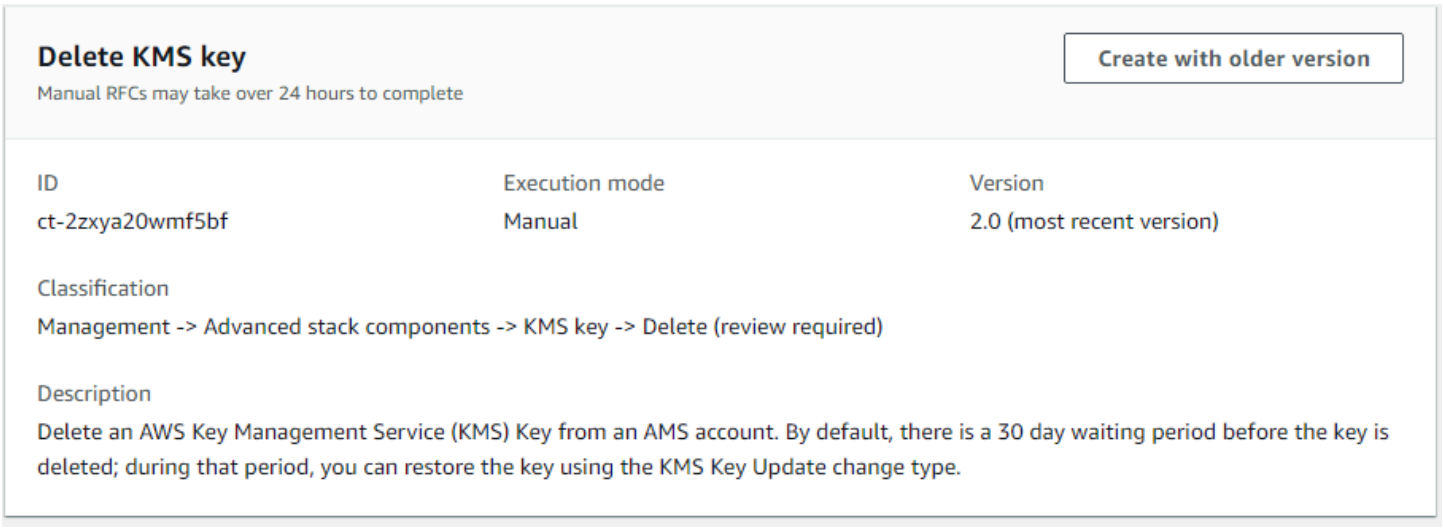
Change type ID	ct-2zxya20wmf5bf
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete KMS key (review required)

Deleting an AWS KMS Key (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AWS KMS Key (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2zxya20wmf5bf" --change-type-version "2.0" --
title "TITLE" --execution-parameters "{\"KeyName\": \"example-kms-key\", \"Operation\":
 \"Delete\",\"KeyDeletionWaitPeriod\":30}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteKmsKeyParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2zxya20wmf5bf" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteKmsKeyParams.json
   ```

2. Modify and save the DeleteKmsKeyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "KeyName": "example-kms-key",
     "Operation": "Delete",
     "KeyDeletionWaitPeriod": 30
   }
   ```

3. Output the RFC template JSON file to a file; this example names it DeleteKmsKeyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteKmsKeyRfc.json
   ```

4. Modify and save the DeleteKmsKeyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "2.0",
   "ChangeTypeId":         "ct-2zxya20wmf5bf",
   "Title":                "KmsKey-Delete-RFC"
   }
   ```

5. Create the RFC, specifying the DeleteKmsKey Rfc file and the DeleteKmsKeyParams file:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteKmsKeyRfc.json  --execution-
   parameters file://DeleteKmsKeyParams.json
   ```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

- To learn more about deleting KMS keys, see Deleting AWS KMS keys.
- This change type has moved to v2.0 with the addition of a new parameter, KeyDeletionWaitPeriod, that you use to set a delay in the key deletion, 7 - 30 days (30 days is the default).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2zxya20wmf5bf.

Example: Required Parameters

```
{
  "KeyName": "kms_key_name",
  "Operation": "Delete",
  "KeyDeletionWaitPeriod": 30
}
```

Example: All Parameters

```
{
  "KeyName": "kms_key_name",
  "Operation": "Delete",
  "KeyDeletionWaitPeriod": 30,
  "Priority": "Medium"
}
```

KMS Key | Enable Rotation

Enable automatic key rotation for an AWS Key Management Service (KMS) customer master key (CMK).

Full classification: Management | Advanced stack components | KMS key | Enable rotation

Change Type Details

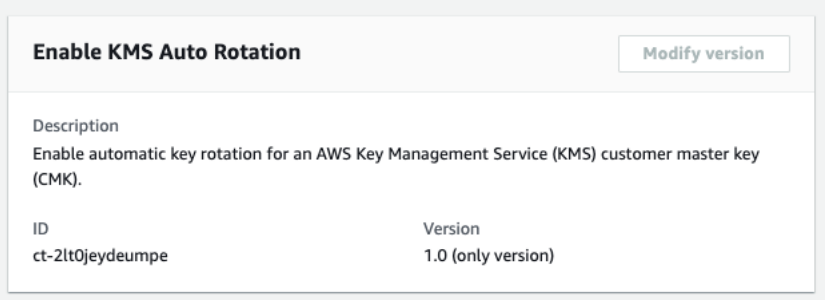
Change type ID	ct-2lt0jeydeumpe
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable automatic KMS key rotation

Enabling auto rotation for a AWS KMS key with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling auto rotation for a AWS KMS key with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title kms-key-enable-rotation --change-type-id
 ct-2lt0jeydeumpe --change-type-version 1.0 --execution-parameters '{"DocumentName":
 "AWSManagedServices-EnableKMSKeyRotation", "Region": "us-east-1", "Parameters":
 {"KeyId": ["12345678-90ab-cdef-1234-567890abcdef"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it KmsKeyEnableRotationParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2lt0jeydeumpe"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    KmsKeyEnableRotationParams.json
   ```

2. Modify and save the KmsKeyEnableRotationParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-EnableKMSKeyRotation",
     "Region": "us-east-1",
     "Parameters": {
       "KeyId": [
         "12345678-90ab-cdef-1234-567890abcdef"
       ]
   ```

```
  }
}
```

3. Output the RFC template JSON file to a file; this example names it KmsKeyEnableRotationRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > KmsKeyEnableRotationRfc.json
```

4. Modify and save the KmsKeyEnableRotationRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2lt0jeydeumpe",
  "Title": "enable-kms-rotation"
}
```

5. Create the RFC, specifying the KmsKeyEnableRotation Rfc file and the KmsKeyEnableRotationParams file:

```
aws amscm create-rfc --cli-input-json file://KmsKeyEnableRotationRfc.json  --
execution-parameters file://KmsKeyEnableRotationParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about KMS, see Key Management Service.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2lt0jeydeumpe.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-EnableKMSKeyRotation",
  "Region": "us-east-1",
  "Parameters": {
```

```
    "KeyId": [
      "58c399bf-1662-4d55-8bbe-fb6d26bd72b9"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-EnableKMSKeyRotation",
  "Region": "us-east-1",
  "Parameters": {
    "KeyId": [
      "arn:aws:kms:us-east-1:123456789012:key/58c399bf-1662-4d55-8bbe-fb6d26bd72b9"
    ]
  }
}
```

KMS Key | Share (Review Required)

Allow cross-account access to a KMS key by adding a statement to the key policy with encrypt and decrypt permissions.

Full classification: Management | Advanced stack components | KMS key | Share (review required)

Change Type Details

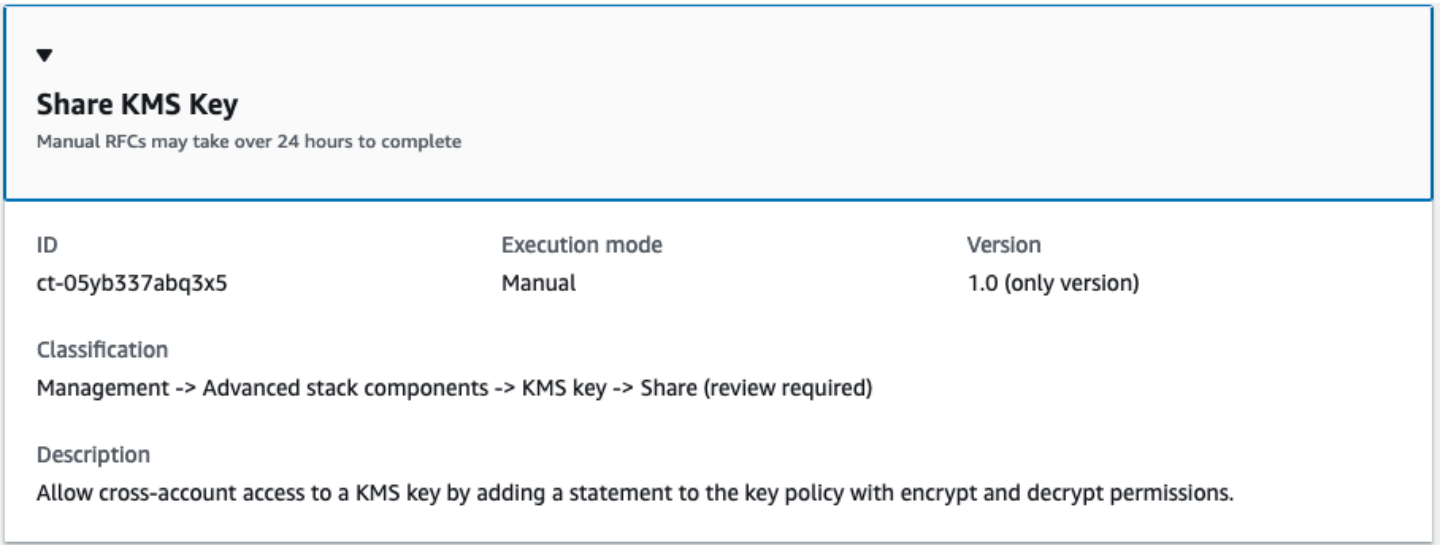
Change type ID	ct-05yb337abq3x5
Current version	1.0
Expected execution duration	30 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Share AWS KMS Key

Share an AWS KMS key with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Share an AWS KMS key with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title="Add Static Route" --description="Share KMS Key"
 --ct-id="ct-05yb337abq3x5" --ct-version="1.0" --input-params="{\"KMSKeyArn\":
\"arn:aws:kms:us-east-1:111122223333:key/06506094-64e2-47f3-94bd-f919eefa22f5\",
\"TargetAccountId\":\"000000000000\",\"IncludeKeyGrantOperations\":\"false\",
\"IAMUserOrRole\":\"arn:aws:iam::000000000000:role/role-name\", \"Priority\":\"High\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ShareKmsKeyParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-05yb337abq3x5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > ShareKmsKeyParams.json
   ```

   Modify and save the ShareKmsKeyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "Share KMS Key",
     "Parameters": {
       "KMSKeyArn": "arn:aws:kms:us-east-1:111122223333:key/06506094-64e2-47f3-94bd-
   f919eefa22f5",
       "TargetAccountId": "000000000000",
       "IncludeKeyGrantOperations": "false"
       "IAMUserOrRole": "arn:aws:iam::000000000000:role/role-name"
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it ShareKmsKeyParamsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ShareKmsKeyParamsRfc.json
```

3. Modify and save the ShareKmsKeyParams.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId": {
  "ChangeTypeVersion":    "1.0",
  "ChangeTypeId":         "ct-05yb337abq3x5",
  "Title":                "Share KMS Key"
}
```

4. Create the RFC, specifying the ShareKmsKeyParamsRfc file and the ShareKmsKeyParams file:

```
aws amscm create-rfc --cli-input-json file://ShareKmsKeyParamsRfc.json --execution-
parameters file://ShareKmsKeyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

To log in to the instance through a bastion, follow the next procedure, Instance access examples.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-05yb337abq3x5.

Example: Required Parameters

```
{
  "KMSKeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "TargetAccountId": "123456789012"
}
```

Example: All Parameters

```
{
```

```
  "KMSKeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "IncludeKeyGrantPermissions": true,
  "TargetAccountId": "111122223333",
  "IAMUserOrRoleARN": "arn:aws:iam::123456789012:role/my_role",
  "Priority": "Medium"
}
```

KMS Key | Update (Review Required)

Request an update of a KMS Key.

Full classification: Management | Advanced stack components | KMS key | Update (review required)

Change Type Details

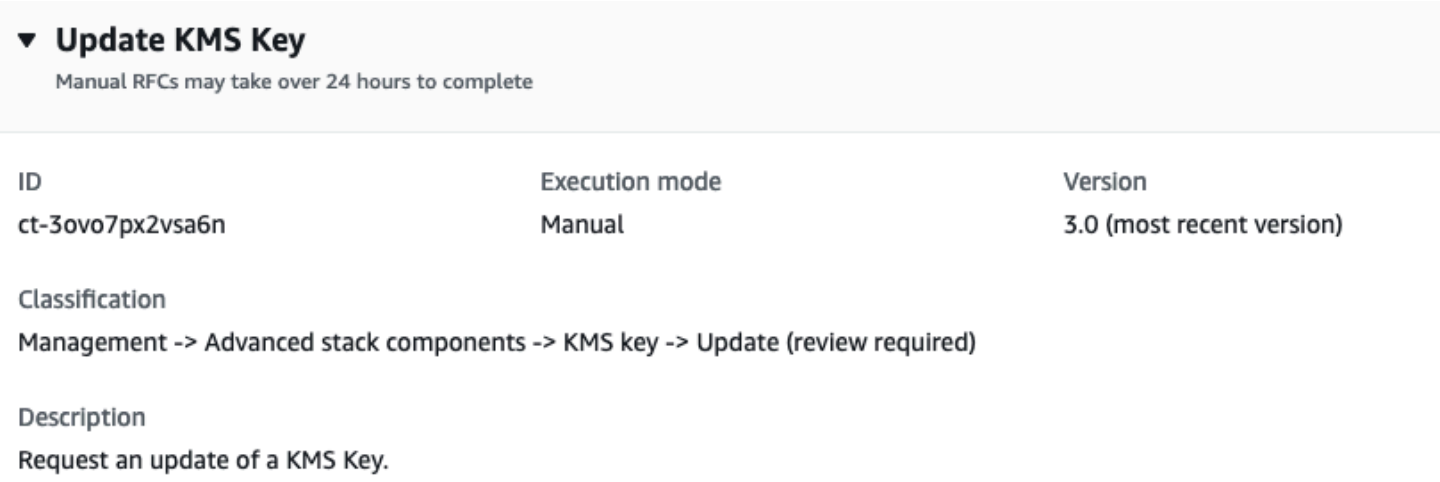
Change type ID	ct-3ovo7px2vsa6n
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update KMS key (review required)

Updating an AWS KMS Key (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AWS KMS Key (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3ovo7px2vsa6n" --change-type-version "3.0"
 --title "TITLE" --execution-parameters "{\"KeyDescription\": \"Example description\",
 \"KeyPermissions\": \"key permissions\", \"PolicyAction\": \"Replace\", \"Operation\":
 \"Update\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateKmsKeyParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3ovo7px2vsa6n" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateKmsKeyParams.json
   ```

2. Modify and save the UpdateKmsKeyParams file. For example, you can replace the contents with something like this:

   ```
   {
   "KeyDescription": "KMS key request",
   "PolicyAction": "Replace",
   "KeyPermissions": "{\"Id\":\"key-consolepolicy-3\",\"Version\":\"2012-10-17\",
   \"Statement\":[{\"Sid\":\"Allow use of the key\",\"Effect\":\"Allow\",\"Principal
   \":{\"AWS\":[\"arn:aws:iam::111122223333:role/KMSRole\"]},\"Action\":[\"kms:Encrypt
   \",\"kms:Decrypt\",\"kms:ReEncrypt*\",\"kms:GenerateDataKey*\",\"kms:DescribeKey
   \"],\"Resource\":\"*\"}]}",
   "Operation": "Update"
   }
   ```

3. Output the RFC template JSON file to a file; this example names it UpdateKmsKeyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateKmsKeyRfc.json
   ```

4. Modify and save the UpdateKmsKeyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "3.0",
   "ChangeTypeId":         "ct-3ovo7px2vsa6n",
   "Title":                "KmsKey-Update-RFC"
   }
   ```

5. Create the RFC, specifying the UpdateKmsKey Rfc file and the UpdateKmsKeyParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateKmsKeyRfc.json  --execution-
parameters file://UpdateKmsKeyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type has moved to v2.0 with the addition of new parameter options for the KeyStatus parameter. You can now choose to cancel a KMS key deletion operation and enable or disable the key.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

To learn more about AWS KMS keys, see AWS Key Management Service (KMS), AWS Key Management Service FAQs, and AWS Key Management Service Concepts.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3ovo7px2vsa6n.

Example: Required Parameters

```
{
  "TargetKeyARN": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "Operation": "Update"
}
```

Example: All Parameters

```
{
  "KeyDescription": "Exmample description of the key to be created.",
  "TargetKeyARN": "arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "AliasName": "kms_key_name",
  "KeyStatus": "Enabled",
  "KeyRotation": true,
  "KeyPermissions": "KMS Key permissions to add: kms:Get",
  "Tags": [
    {
      "Key": "foo",
      "Value": "bar"
    },
    {
      "Key": "testkey",
      "Value": "testvalue"
    }
  ],
  "Operation": "Update",
  "Priority": "Medium"
}
```

Load Balancer (ELB) Stack | Replace Listener Certificate

Replace the certificate of an existing Elastic (Classic) Load Balancer (ELB) listener. Use the RemediateDrift parameter to have the automation try to remediate the stack drift, if drift is introduced in the CloudFormation stack that was used to create the load balancer.

Full classification: Management | Advanced stack components | Load balancer (ELB) stack | Replace listener certificate

Change Type Details

Change type ID	ct-0aqx5t0pgfzbg
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required

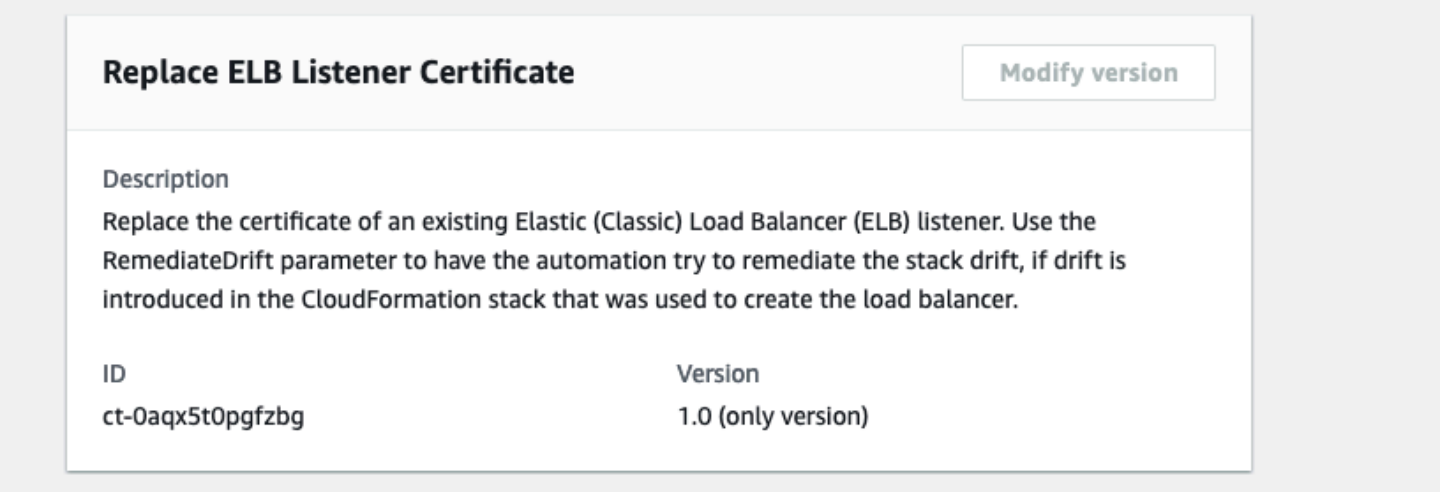
Customer approval	Not required
Execution mode	Automated

Additional Information

Replace an ELB listener certificate

Replacing an ELB listener certificate with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Replacing an ELB listener certificate with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-0aqx5t0pgfzbg" --change-type-version
 "1.0" --title "Replace listener certificate" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-SetClassicLoadBalancerCertificate\",\"Region\": \"us-
east-1\",\"Parameters\":{\"LoadBalancerName\":[\"testalb\"],\"SSLCertificateArn
\":[\"arn:aws:acm:us-east-1:123456789012:certificate/c96c73cd-d082-4fa9-
bbf2-09d8600d84ad\"],"LoadBalancerPort":[\"443\"],\"RemediateStackDrift\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it ReplaceListCertParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0aqx5t0pgfzbg"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ReplaceListCertParams.json
   ```

2. Modify and save the ReplaceListCertParams file. The values given in the example reflect a deployment of a Public ELB, with the health check thresholds relaxed and the ELBScheme set to `true` (for a public ELB). Note that the `Name` you set here is not the actual ELB name, you can find that name in the console as the ELB instance name. Not all optional parameters are shown in the example.

   ```
   {
       "DocumentName": "AWSManagedServices-SetClassicLoadBalancerCertificate",
       "Region": "us-east-1",
       "Parameters": {
           "LoadBalancerName": [
               "testalb"
   ```

```
        ],
        "SSLCertificateArn": [
            "arn:aws:acm:us-east-1:123456789012:certificate/c96c73cd-d082-4fa9-
bbf2-09d8600d84ad"
        ],
        "LoadBalancerPort":[
             "443"
         ]
        "RemediateStackDrift": [
            "True"
        ]
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it ReplaceListCertRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ReplaceListCertRfc.json
```

4. Modify and save the ReplaceListCertRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0aqx5t0pgfzbg",
"Title":                "My-ELB-Create-RFC"
}
```

5. Create the RFC, specifying the ReplaceListCertRfc file and the ReplaceListCertParams file:

```
aws amscm create-rfc --cli-input-json file://ReplaceListCertRfc.json --execution-
parameters file://ReplaceListCertParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the load balancer, look in the execution output: Use the `stack_id` to view the ELB in the Cloud Formation console or to create a Delete Stack RFC, use the ELBCName value to programmatically access the ELB.

Tips

For information about Application Load Balancers, see Application Load Balancers.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0aqx5t0pgfzbg.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-SetClassicLoadBalancerCertificate",
    "Region": "us-east-1",
    "Parameters": {
        "LoadBalancerName": [
            "testclassiclb"
        ],
        "SSLCertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-SetClassicLoadBalancerCertificate",
    "Region": "us-east-1",
    "Parameters": {
        "LoadBalancerName": [
            "testclassiclb"
        ],
        "LoadBalancerPort": [
            "443"
        ],
        "RemediateStackDrift": [
            "False"
        ],
        "SSLCertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
```

```
        ]
    }
}
```

Load Balancer (ELB) Stack | Update

Modify the properties of an existing Amazon ELB Classic Load Balancer created using CT id ct-12amsdz909cfh, version 3.0.

Full classification: Management | Advanced stack components | Load balancer (ELB) stack | Update

Change Type Details

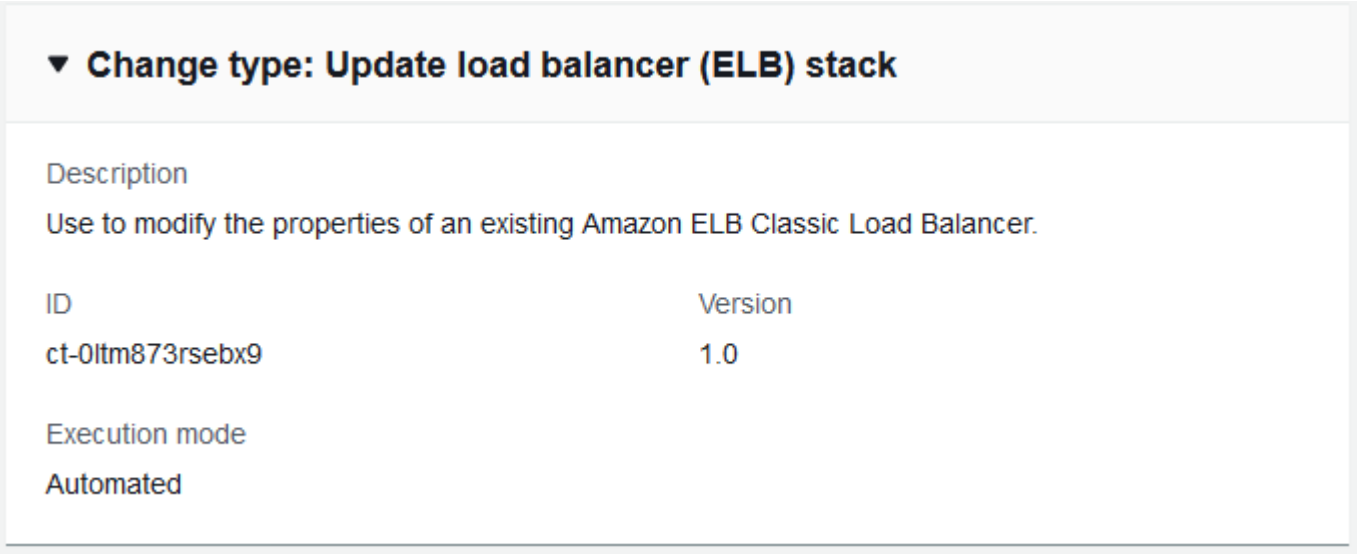
Change type ID	ct-0ltm873rsebx9
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update ELB load balancer

Updating an Elastic Load Balancer with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an Elastic Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title my-db-instance --change-type-id
 ct-0ltm873rsebx9 --change-type-version 3.0 --execution-parameters
 '{"VpcId":"VPC_ID","StackId":"STACK_ID","Parameters":{"ELBBackendInstances":
["INSTANCE_ID1","INSTANCE_ID2"], "ELBIdleTimeout": "600"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateElbParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ltm873rsebx9" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateElbParams.json
   ```

2. Modify and save the UpdateElbParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "ELB-Update",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-sdhopv00000000000",
   "Name":             "My-ELB",

   "Parameters":   {
       "ELBSubnetIds":  ["PUBLIC_AZ1", "PUBLIC_AZ2"],
       "ELBHealthCheckHealthyThreshold":   2,
       "ELBHealthCheckInterval":           30,
       "ELBHealthCheckTarget":             "HTTP:80/status",
       "ELBHealthCheckTimeout":            10,
       "ELBHealthCheckUnhealthyThreshold": 3,
       "ELBScheme":                        true
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdateElbRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateElbRfc.json
   ```

4. Modify and save the UpdateElbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-0ltm873rsebx9",
"Title":                "ELB-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateElbRfc file and the UpdateElbParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateElbRfc.json --execution-
parameters file://UpdateElbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. You might need to submit a Management | Other | Other | Update change type to open ports and associate security groups, see Other | Other requests.

Tips

To learn more about AWS Classic Load Balancers, see What Is a Classic Load Balancer?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ltm873rsebx9.

Example: Required Parameters

```
{
  "VpcId": "vpc-01234567890abcdef",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
  }
}
```

Example: All Parameters

```
{
```

```
  "VpcId": "vpc-01234567890abcdef",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "ELBSubnetIds": ["subnet-a0b1c2d3", "subnet-a0b2c9d8"],
    "ELBHealthCheckHealthyThreshold": 2,
    "ELBHealthCheckInterval": 10,
    "ELBHealthCheckTarget": "HTTP:80/index.html",
    "ELBHealthCheckTimeout": 10,
    "ELBHealthCheckUnhealthyThreshold": 3,
    "ELBIdleTimeout": 30,
    "ELBInstancePort": "80",
    "ELBInstanceProtocol": "HTTPS",
    "ELBCookieExpirationPeriod": "60",
    "ELBCookieStickinessPolicyName": "MyPolicy",
    "ELBLoadBalancerPort": "443",
    "ELBLoadBalancerProtocol": "HTTP",
    "ELBSSLCertificateId": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012",
    "ELBCrossZone": true,
    "ELBBackendInstances": ["i-1234567a", "i-1234567b"],
    "ELBInstancePort2": "80",
    "ELBInstanceProtocol2": "HTTPS",
    "ELBCookieExpirationPeriod2": "60",
    "ELBCookieStickinessPolicyName2": "MyPolicy2",
    "ELBLoadBalancerPort2": "445",
    "ELBLoadBalancerProtocol2": "HTTP",
    "ELBSSLCertificateId2": "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
  }
}
```

Network Load Balancer | Add Listener Certificate

Add a certificate to the specified Network Load Balancer (NLB) listener. Use the RemediateStackDrift parameter for the automation to try to remediate drift, if it is introduced.

Full classification: Management | Advanced stack components | Network Load Balancer | Add listener certificate

Change Type Details

Change type ID	ct-35p977vul06df

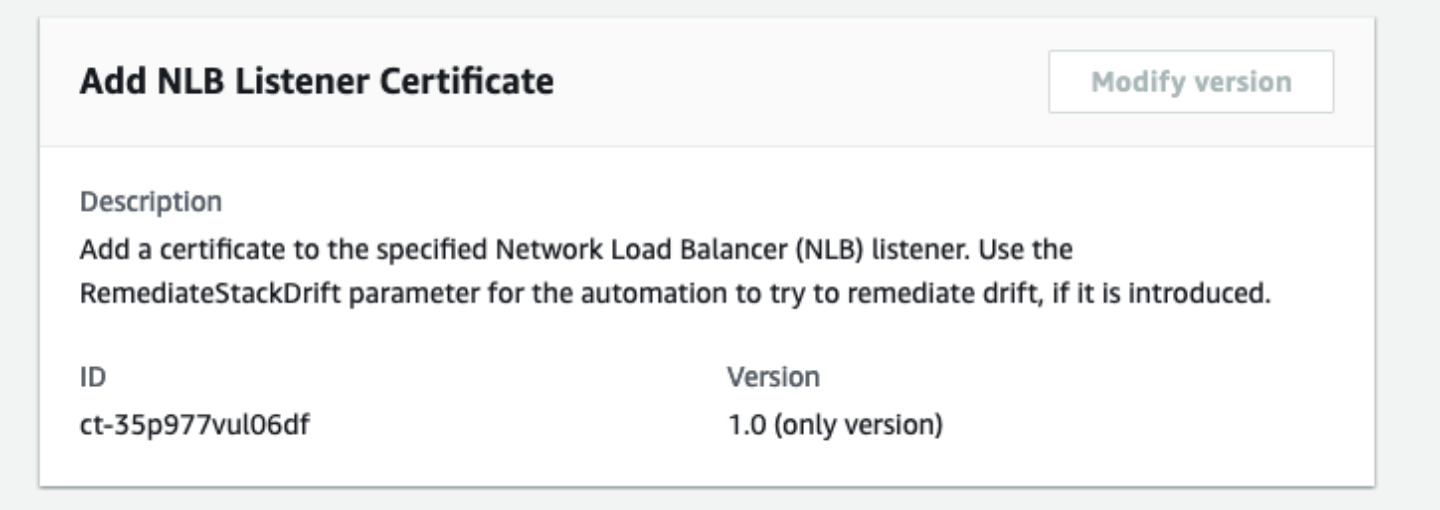
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add NLB listener certificate

Adding a listener certificate to an NLB with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a listener certificate to an NLB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-35p977vul06df" --change-type-version
 "1.0" --title "Add listener certificate NLB" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-AddCertificateToElbv2Listener\",\"Region\": \"us-
east-1\",\"Parameters\":{\"ListenerArn\":[\"arn:aws:elasticloadbalancing:us-
east-1:123456789012:listener/app/testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e\"],
\"CertificateArn\":[\"arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba\"],\"IsDefault\":[\"False\"],\"RemediateStackDrift
\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

   ```
   aws amscm get-change-type-version --change-type-id "ct-35p977vul06df"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AddNlbListenerCertParams.json
   ```

2. Modify and save the AddNlbListenerCertParams file. For example:

   ```
   {
       "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
       "Region": "us-east-1",
       "Parameters": {
   ```

```
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e"
        ],
        "CertificateArn": [
            "arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba"
        ],
        "IsDefault": [
            "False"
        ],
        "RemediateStackDrift": [
            "True"
        ]
    }
}
```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > AddNlbListenerCertRfc.json
```

4. Modify and save the AddNlbListenerCertRfc.json file. For example:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-35p977vul06df",
"Title":                "NLB-Add-Listener-Cert-RFC"
}
```

5. Create the RFC, specifying the AddNlbListenerCertRfc file and the AddNlbListenerCertParams file:

```
aws amscm create-rfc --cli-input-json file://AddNlbListenerCertRfc.json --
execution-parameters file://AddNlbListenerCertParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Network Load Balancers, see Create a Network Load Balancer.

To create a network load balancer listener, see Target Group | Create (For NLB).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-35p977vul06df.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/net/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-AddCertificateToElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "IsDefault": [
            "True"
        ],
        "ListenerArn": [
```

```
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/net/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ],
        "RemediateStackDrift": [
            "False"
        ]
    }
}
```

Network Load Balancer | Remove Listener Certificate

Remove a certificate from the specified Network Load Balancer (NLB) listener. Use the RemediateStackDrift parameter for the automation to try to remediate drift, if it is introduced.

Full classification: Management | Advanced stack components | Network Load Balancer | Remove listener certificate

Change Type Details

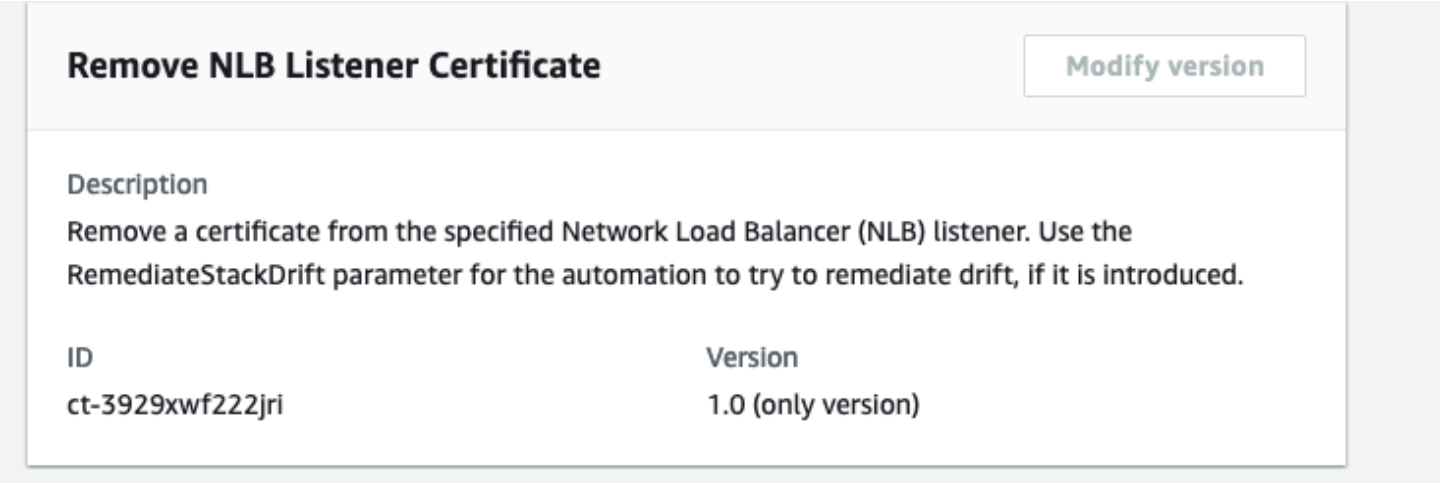
Change type ID	ct-3929xwf222jri
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove NLB listener certificate

Removing a listener certificate from an NLB with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing a listener certificate from an NLB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-3929xwf222jri" --change-type-version "1.0"
 --title "Remove listener certificate NLB" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-RemoveCertificateToElbv2Listener\",\"Region\": \"us-
east-1\",\"Parameters\":{\"ListenerArn\":[\"arn:aws:elasticloadbalancing:us-
east-1:123456789012:listener/app/testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e\"],
\"CertificateArn\":[\"arn:aws:acm:us-east-1:123456789012:certificate/
ecb242e8-3da5-4da6-813c-17040f086fba\"],\"IsDefault\":[\"False\"],\"RemediateStackDrift
\":[\"True\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file. For example, you can replace the contents with something like this:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3929xwf222jri"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RemoveNlbListenerCertParams.json
   ```

2. Modify and save the RemoveNlbListenerCertParams file. For example:

   ```
   {
       "DocumentName": "AWSManagedServices-RemoveCertificateToElbv2Listener",
       "Region": "us-east-1",
       "Parameters": {
           "ListenerArn": [
               "arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
   testalb/fc656bcb5cacb3ae/a0c0da77f9b1461e"
           ],
           "CertificateArn": [
               "arn:aws:acm:us-east-1:123456789012:certificate/
   ecb242e8-3da5-4da6-813c-17040f086fba"
           ],
           "IsDefault": [
               "False"
           ],
           "RemediateStackDrift": [
               "True"
           ]
       }
   }
   ```

3. Output the RFC template to a file in your current folder. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --generate-cli-skeleton > RemoveNlbListenerCertRfc.json
```

4. Modify and save the RemoveNlbListenerCertRfc.json file. For example:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3929xwf222jri",
"Title":                "NLB-Remove-Listener-Cert-RFC"
}
```

5. Create the RFC, specifying the RemoveNlbListenerCertRfc file and the RemoveNlbListenerCertParams file:

```
aws amscm create-rfc --cli-input-json file://RemoveNlbListenerCertRfc.json --
execution-parameters file://RemoveNlbListenerCertParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about listeners, see ELB Listeners.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3929xwf222jri.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-RemoveCertificateFromElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
```

```
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/net/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-RemoveCertificateFromElbv2Listener",
    "Region": "us-east-1",
    "Parameters": {
        "CertificateArn": [
            "arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012"
        ],
        "ListenerArn": [
            "arn:aws:elasticloadbalancing:us-west-2:123456789012:listener/net/my-load-
balancer/50dc6c495c0c9188/50dc6c495c0c9188"
        ],
        "RemediateStackDrift": [
            "False"
        ]
    }
}
```

Network Load Balancer | Update

Update the properties of an existing Network Load Balancer.

Full classification: Management | Advanced stack components | Network Load Balancer | Update

Change Type Details

Change type ID	ct-0wglhholzo0uw
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required

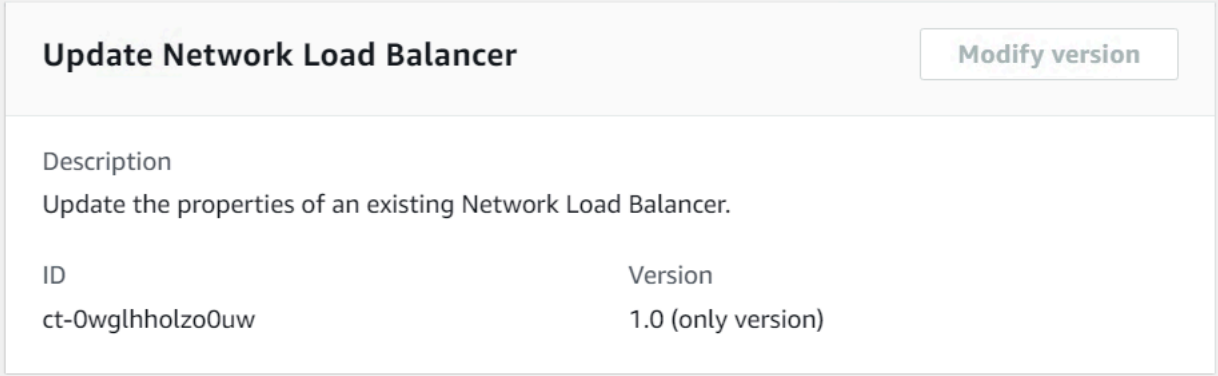
Customer approval	Not required
Execution mode	Automated

Additional Information

Update NLB load balancer

Updating an NLB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an NLB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title test-update-nlb --change-type-id
 ct-0wglhholzo0uw --change-type-version 1.0 --execution-parameters
 '{"Description":"Update NLB","VpcId":"vpc-1234abcd","StackTemplateId":"stm-
l70qr9itukvqssg8d","Name":"test-update-nlb","TimeoutInMinutes":60,"Parameters":
{"HealthCheckHealthyThreshold": 4,"HealthCheckIntervalSeconds": 20,"HealthCheckTargetPath":
 "/","HealthCheckTargetPort": 80,"HealthCheckTargetProtocol":
 "TCP","CrossZoneEnabled": false,"ProxyProtocolV2": false,"DeregistrationDelayTimeoutSeconds":
 "i-123456789abcdefgh","Target1Port": 80, "Target1AvailabilityZone": "AZ"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateNlbParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0wglhholzo0uw" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateNlbParams.json
   ```

2. Modify and save the UpdateNlbParams file. The values given in the example reflect a deployment of a Public NLB, with the health check thresholds relaxed and the `Public` parameters set to true (for a public NLB). Note that the `Name` you set here is not the actual NLB name, you can find that name in the console as the NLB instance name.

   ```
   {
   "Description":      "NLB-Create",
   "VpcId":            "VPC_ID",
   "StackTemplateId":  "stm-l70qr9itukvqssg8d",
   "Name":             "My-NLB",

   "Parameters":   {
       "SubnetIds":  ["PUBLIC_AZ1", "PUBLIC_AZ2"],
       "HealthCheckHealthyThreshold":   2,
   ```

```
    "HealthCheckInterval":           30,
    "HealthCheckTargetPath":         traffic-port",
    "DeregistrationDelayTimeout":                10,
    "Public":                        true
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateNlbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateNlbRfc.json
```

4. Modify and save the UpdateNlbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0wglhholzo0uw",
"Title":                "NLB-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateNlbRfc file and the UpdateNlbParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateNlbRfc.json --execution-
parameters file://UpdateNlbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You can specify up to four Target IDs, Ports, and Availability Zones.

To learn more about AWS Network Load Balancers, see Create a Network Load Balancer.

To create a network load balancer listener, see Target Group | Create (For NLB).

To create a network load balancer target group, see Create NLB target group.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0wglhholzo0uw.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-1234567890abcdef0",
  "Parameters": {
    "HealthCheckHealthyThreshold": "4",
    "HealthCheckIntervalSeconds": "10",
    "HealthCheckTargetPath": "/",
    "HealthCheckTargetPort": "80",
    "HealthCheckTargetProtocol": "TCP",
    "CrossZoneEnabled": "false",
    "ProxyProtocolV2": "false",
    "DeregistrationDelayTimeoutSeconds": "360",
    "Target1ID": "i-123456789abcdefgh",
    "Target1Port": "80",
    "Target1AvailabilityZone": "us-east-1a"
  }
}
```

RDS Database Stack | Reboot

Use to reboot an RDS DB instance.

Full classification: Management | Advanced stack components | RDS database stack | Reboot

Change Type Details

Change type ID	ct-0bpxsrtu16igp

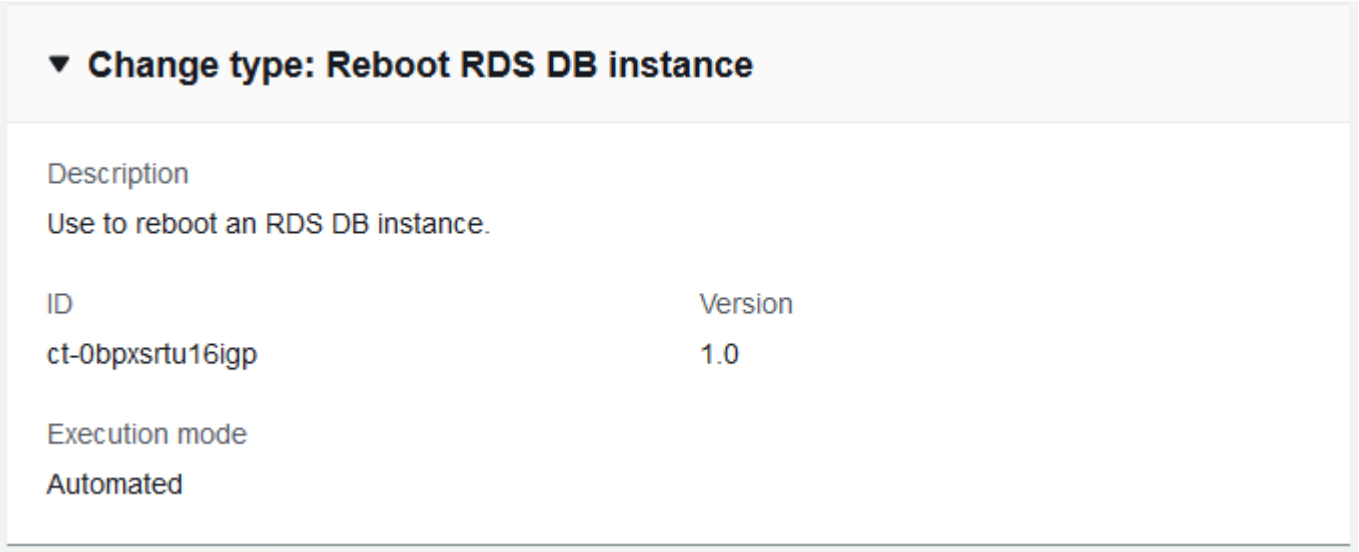
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Reboot DB stack

Rebooting an RDS Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Rebooting an RDS Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0bpxsrtu16igp" --change-type-version "1.0"
 --title "RDS-Reboot" --execution-parameters "{\"DbInstanceIdentifier\":\"DB_ID\",
\"ForceFailover\":false}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named RebootRdsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0bpxsrtu16igp" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RebootRdsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DbInstanceIdentifier": "DB_ID",
     "ForceFailover": true
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it RebootRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RebootRdsRfc.json
```

4. Modify and save the RebootRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0bpxsrtu16igp",
"Title":                "RDS-Reboot-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the RebootRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://RebootRdsRfc.json --execution-
parameters file://RebootRdsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about RDS, see the RDS User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0bpxsrtu16igp.

Example: Required Parameters

```
{
  "DbInstanceIdentifier": "dbinstance"
}
```

Example: All Parameters

```
{
```

```
  "DbInstanceIdentifier": "db-instance1",
  "ForceFailover": true
}
```

RDS Database Stack | Restore To Point In Time

Restore an RDS DB instance to a point in time.

Full classification: Management | Advanced stack components | RDS database stack | Restore to point in time

Change Type Details

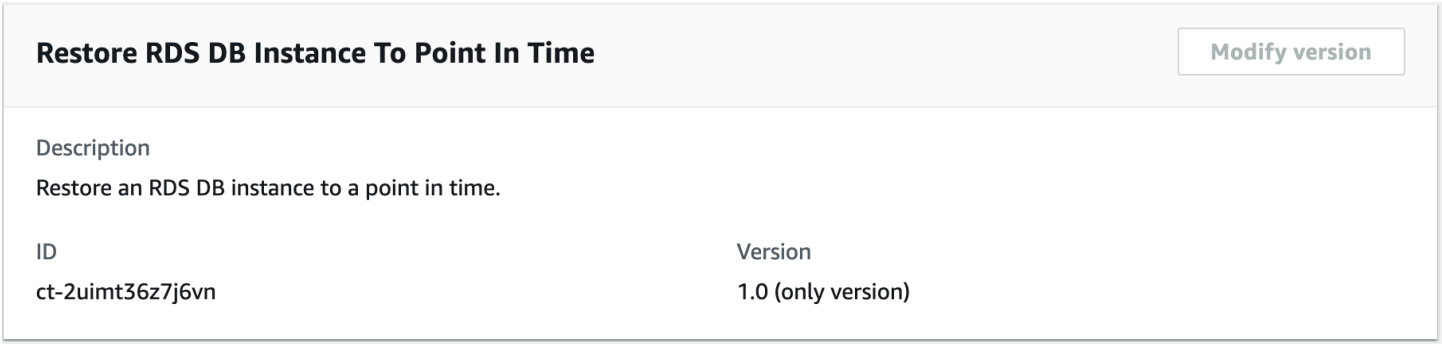
Change type ID	ct-2uimt36z7j6vn
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Restore DB to point in time

Restoring an RDS DB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Restoring an RDS DB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title create-rds-db-instance-point-in-time-restore --
change-type-id ct-2uimt36z7j6vn --change-type-version 1.0 --execution-parameters
 '{"DocumentName": "AWSManagedServices-RestoreRDSInstanceToPointInTime","Region":
 "us-east-1","Parameters": {"SourceDBInstanceIdentifier": ["my-application-
db"],"TargetDBInstanceIdentifier": ["my-application-db-restore"],"RestoreTime":
 ["2021-03-28T00:00:00Z"],"DBInstanceClass": ["db.t3.micro"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named RestoreRdsDbParams.json.

   ```
   aws amscm create-rfc --cli-input-json file://RestoreRdsDbRFC.json --execution-
   parameters file://RestoreRdsDbParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-RestoreRDSInstanceToPointInTime",
  "Region": "us-east-1",
  "Parameters": {
     "SourceDBInstanceIdentifier": [
        "my-application-db"
      ],
     "TargetDBInstanceIdentifier": [
        "my-application-db-restore"
      ],
     "RestoreTime": [
        "2021-03-28T00:00:00Z"
      ],
     "DBInstanceClass": [
        "db.t3.micro"
      ]
   }
}
```

3. Output the JSON template to a file in your current folder; this example names it RestoreRdsDbRFC.json:

```
aws amscm create-rfc --generate-cli-skeleton > RestoreRdsDbRFC.json
```

4. Modify and save the RestoreRdsDbParams.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2uimt36z7j6vn",
"Title":                "Restore RDS DB instance to point in time"
}
```

5. Create the RFC, specifying the execution parameters file and the RestoreRdsDbParams file:

```
aws amscm create-rfc --cli-input-json file://RestoreRdsDbRFC.json --execution-
parameters file://RestoreRdsDbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2uimt36z7j6vn.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-RestoreRDSInstanceToPointInTime",
  "Region" : "us-east-1",
  "Parameters" : {
    "SourceDBInstanceIdentifier" : [
      "source-db-instance"
    ],
    "TargetDBInstanceIdentifier" : [
      "restored-db-instance"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-RestoreRDSInstanceToPointInTime",
  "Region" : "us-east-1",
  "Parameters" : {
    "SourceDBInstanceIdentifier" : [
      "source-db-instance"
    ],
    "TargetDBInstanceIdentifier" : [
      "restored-db-instance"
    ],
    "RestoreTime" : [
      "2009-09-07T23:45:00Z"
    ],
    "DBInstanceClass" : [
      "db.m5.xlarge"
```

```
    ],
    "DBOptionGroupName": [
      "default-db-optiongroup"
    ],
    "DBParameterGroupName": [
      "default-db-parameters"
    ]
  }
}
```

RDS Database Stack | Rotate DB Certificate

Rotate the DB certificate on an Amazon Relational Database Service (RDS) database (DB) instance. Update any client applications that use SSL/TLS and the server certificate to connect, to use the new CA certificate beforehand. Not doing this will cause an interruption of connectivity between your applications and your database.

Full classification: Management | Advanced stack components | RDS database stack | Rotate DB certificate

Change Type Details

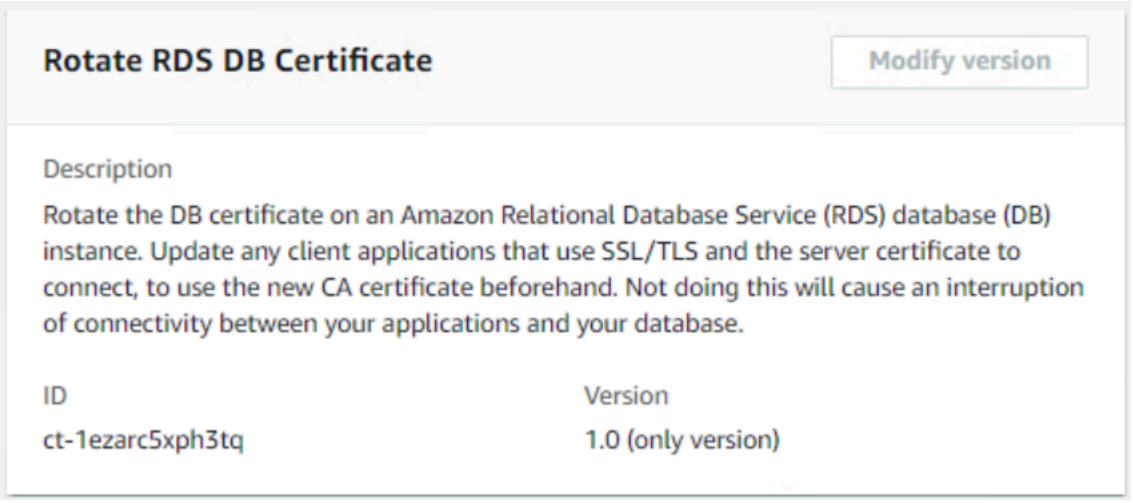
Change type ID	ct-1ezarc5xph3tq
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Rotate DB certificate

Rotating a DB Certificate on an RDS Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Rotating a DB Certificate on an RDS Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1ezarc5xph3tq" --change-type-
version "1.0" --title "Rotate DB Certificate" --execution-parameters
 '{"DocumentName":"AWSManagedServices-RotateDbCertificate","Region":"us-
east-1","Parameters":{"DBInstanceIdentifier":["database-1"],"CertificateIdentifier":
["rds-ca-rsa2048-g1"],"ApplyImmediately":["True"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named RotateRdsCertParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1ezarc5xph3tq" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RotateRdsCertParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "DocumentName": "AWSManagedServices-RotateDbCertificate",
   "Region": "us-east-1",
   "Parameters": {
       "DBInstanceIdentifier": [
         "database-1"
       ],
       "CertificateIdentifier": [
         "rds-ca-rsa2048-g1"
       ],
       "ApplyImmediately": [
         "True"
       ]
   }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it RotateRdsCertRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RotateRdsCertRfc.json
   ```

4. Modify and save the RotateRdsCertRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1ezarc5xph3tq",
"Title":                "RDS-ROTATE-CERT-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the RotateRdsCertRfc file:

```
aws amscm create-rfc --cli-input-json file://RotateRdsCertRfc.json --execution-
parameters file://RotateRdsCertParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> Before scheduling the CA certificate rotation on your database with this change type, update any client applications that use SSL/TLS and the server certificate to connect. Not doing this will cause an interruption of connectivity between your applications and your database.

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1ezarc5xph3tq.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RotateDbCertificate",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": ["dbinstance"],
    "CertificateIdentifier": ["rds-ca-2019"],
    "ApplyImmediately": ["False"]
  }
}
```

RDS Database Stack | Start Aurora Cluster

Start an Aurora DB cluster, which is a provisioned capacity type and does not have cross-region read replicas. The cluster must be in the 'stopped' state.

Full classification: Management | Advanced stack components | RDS database stack | Start Aurora cluster

Change Type Details

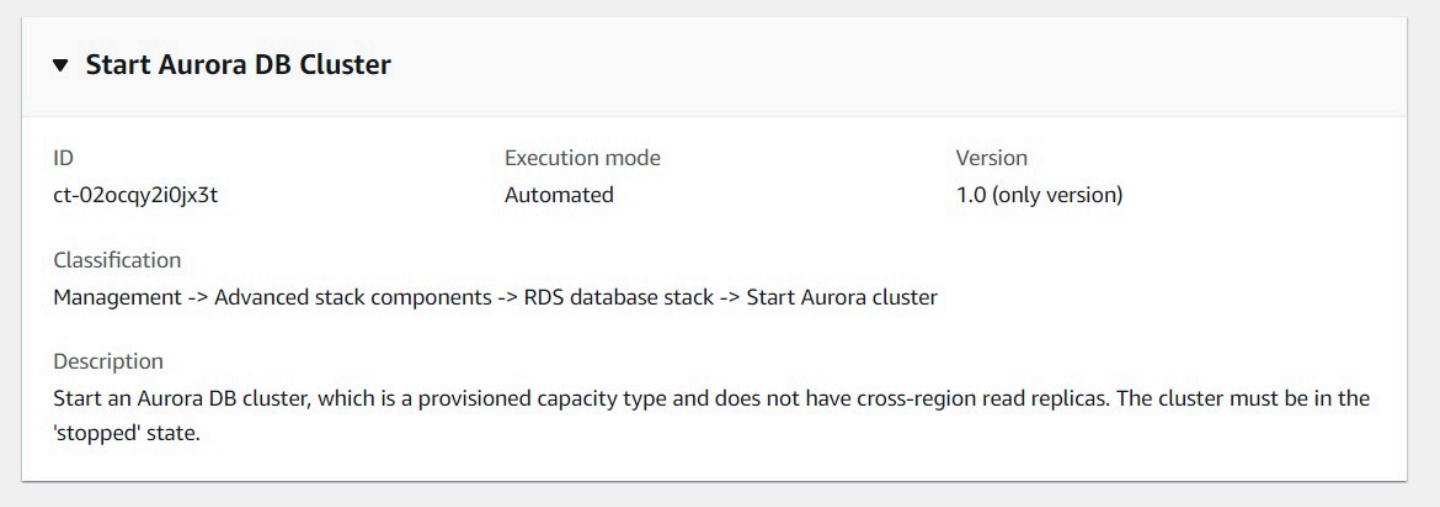
Change type ID	ct-02ocqy2i0jx3t
Current version	1.0
Expected execution duration	90 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start DB Aurora cluster

Starting an RDS DB Aurora cluster with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting an RDS DB Aurora cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-02ocqy2i0jx3t" --change-type-version
 "1.0" --title "Start Aurora DB Cluster" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StartDBCluster\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"DBClusterIdentifier\": \"myaurora-dbcluster\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named StartRdsDbAuroraClusterParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-02ocqy2i0jx3t"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    StartRdsDbAuroraClusterParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-StartDBCluster",
       "Region": "us-east-1",
       "Parameters": {
           "DBClusterIdentifier": "myaurora-dbcluster"
       }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it StartRdsDbAuroraClusterRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StartRdsDbAuroraClusterRfc.json
   ```

4. Modify and save the StartRdsDbAuroraClusterRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
   ```

```
  "ChangeTypeId": "ct-02ocqy2i0jx3t",
  "Title": "Start Aurora DB Cluster"
}
```

5. Create the RFC, specifying the execution parameters file and the StartRdsDbAuroraClusterRfc file:

```
aws amscm create-rfc --cli-input-json file://StartRdsDbAuroraClusterRfc.json --
execution-parameters file://StartRdsDbAuroraClusterParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information see Amazon Aurora DB clusters.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-02ocqy2i0jx3t.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-StartDBCluster",
    "Region": "us-east-1",
    "Parameters": {
        "DBClusterIdentifier": "abcdef01234567890"
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StartDBCluster",
    "Region": "us-east-1",
    "Parameters": {
        "DBClusterIdentifier": "abcdef01234567890"
    }
}
```

RDS Database Stack | Start DB Instance

Start an Amazon Relational Database Service (RDS) database (DB) instance.

Full classification: Management | Advanced stack components | RDS database stack | Start DB instance

Change Type Details

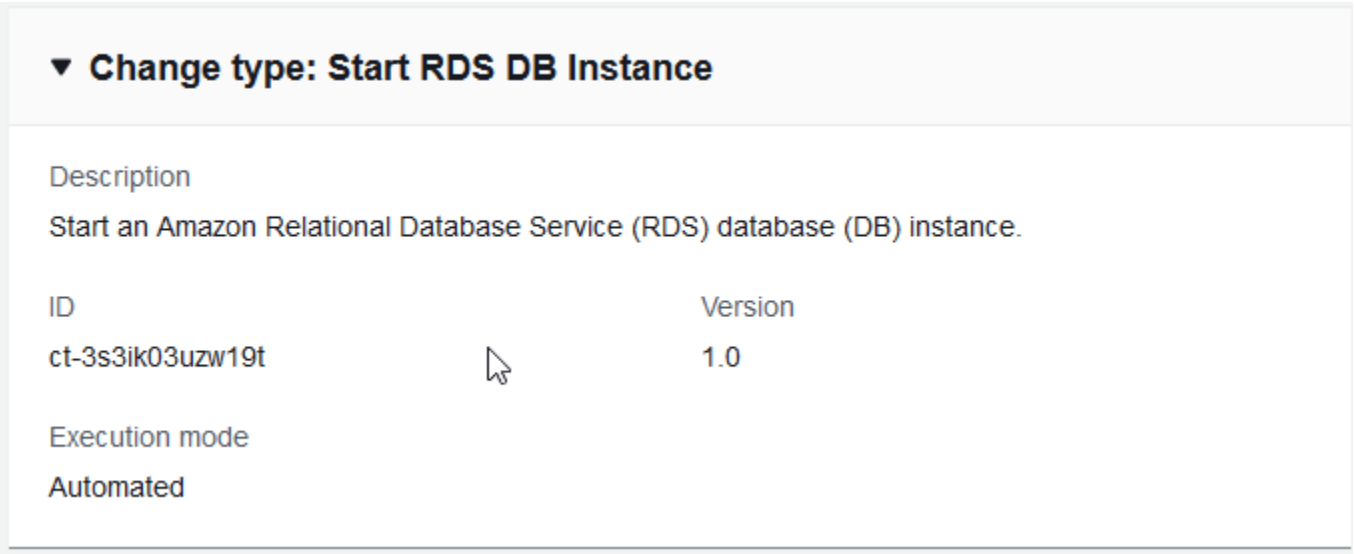
Change type ID	ct-3s3ik03uzw19t
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start DB instance

Starting an RDS DB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting an RDS DB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3s3ik03uzw19t" --change-type-version
 "1.0" --title "Start DB instance" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StartRDSInstance\",\"Region\": \"us-east-1\",\"Parameters\":
{\"InstanceId\":[\"rds-instance\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named StartRdsDbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3s3ik03uzw19t" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StartRdsDbParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
"DocumentName": "AWSManagedServices-StartRDSInstance",
"Region": "us-east-1",
"Parameters": {
  "InstanceId": [ "rds-instance" ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it StartRdsDbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StartRdsDbRfc.json
```

4. Modify and save the StartRdsDbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3s3ik03uzw19t",
"Title":                "RDS-Start-DB-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the StartRdsDbRfc file:

```
aws amscm create-rfc --cli-input-json file://StartRdsDbRfc.json --execution-
parameters file://StartRdsDbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about RDS, see the RDS User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3s3ik03uzw19t.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

RDS Database Stack | Stop Aurora Cluster

Stop an Aurora DB cluster, which is a provisioned capacity type and does not have cross-region read replicas. The cluster must be in the 'available' state.

Full classification: Management | Advanced stack components | RDS database stack | Stop Aurora cluster

Change Type Details

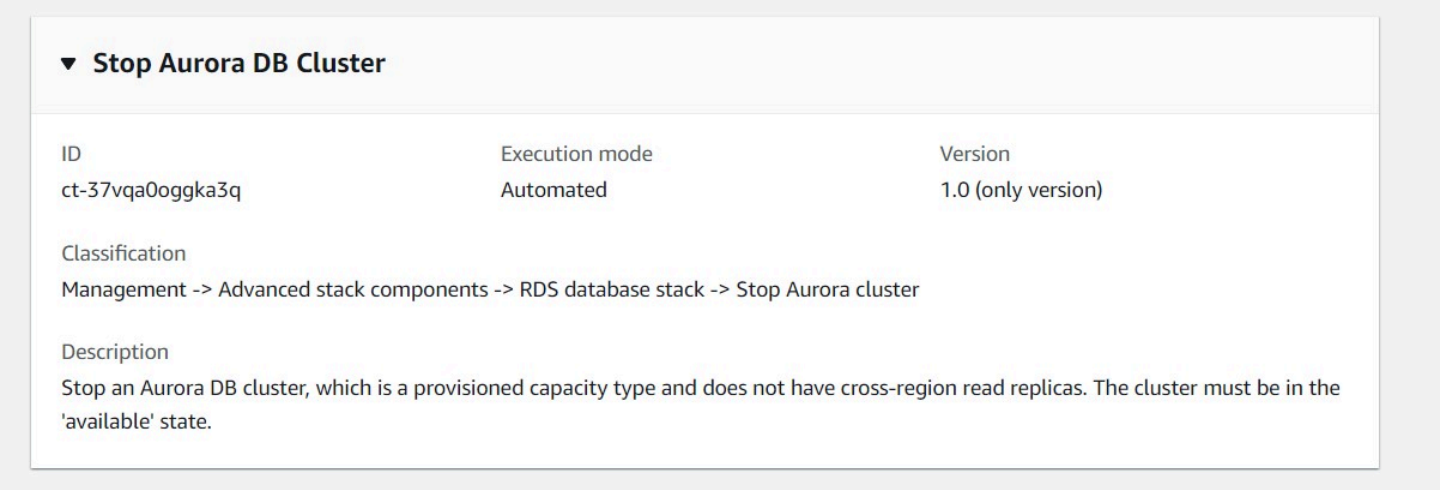
Change type ID	ct-37vqa0oggka3q
Current version	1.0
Expected execution duration	90 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop DB Aurora cluster

Stopping an RDS DB Aurora cluster with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping an RDS DB Aurora cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-37vqa0oggka3q" --change-type-version
 "1.0" --title "Stop Aurora DB Cluster" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StopDBCluster\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"DBClusterIdentifier\": \"myaurora-dbcluster\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named StopRdsDbAuroraClusterParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-37vqa0oggka3q"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    StopRdsDbAuroraClusterParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-StopDBCluster",
       "Region": "us-east-1",
       "Parameters": {
           "DBClusterIdentifier": "myaurora-dbcluster"
       }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it StopRdsDbAuroraClusterRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopRdsDbAuroraClusterRfc.json
   ```

4. Modify and save the StopRdsDbAuroraClusterRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-37vqa0oggka3q",
     "Title": "Stop Aurora DB Cluster"
   }
   ```

5. Create the RFC, specifying the execution parameters file and the StopRdsDbAuroraClusterRfc file:

```
aws amscm create-rfc --cli-input-json file://StopRdsDbAuroraClusterRfc.json --
execution-parameters file://StopRdsDbAuroraClusterParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information see Amazon Aurora DB clusters.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-37vqa0oggka3q.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-StopDBCluster",
    "Region": "us-east-1",
    "Parameters": {
        "DBClusterIdentifier": "abcdef01234567890"
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StopDBCluster",
    "Region": "us-east-1",
    "Parameters": {
        "DBClusterIdentifier": "abcdef01234567890"
    }
}
```

RDS Database Stack | Stop DB Instance

Stop an Amazon Relational Database Service (RDS) database (DB) instance. After seven days, the DB instance is automatically re-started. Supported engines are: MariaDB, Microsoft SQL

Server, MySQL, Oracle, PostgreSQL. This change type doesn't apply to Aurora MySQL and Aurora PostgreSQL.

Full classification: Management | Advanced stack components | RDS database stack | Stop DB instance

Change Type Details

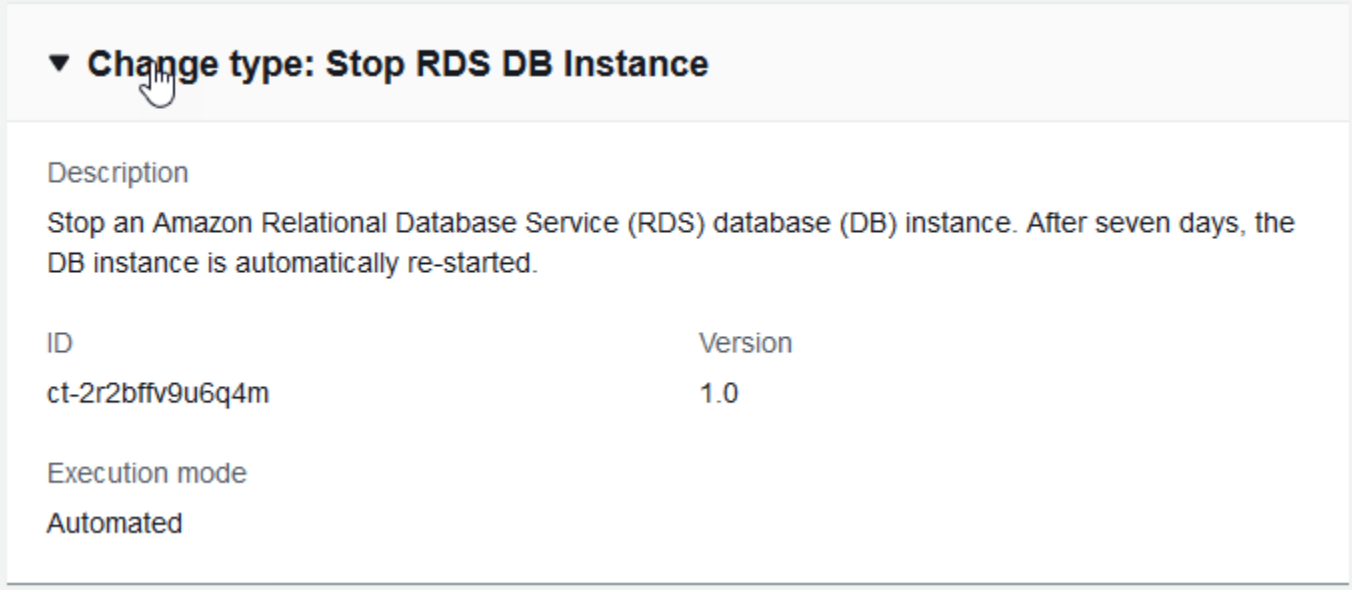
Change type ID	ct-2r2bffv9u6q4m
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop DB instance

Stopping an RDS DB with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping an RDS DB with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2r2bffv9u6q4m" --change-type-version
 "1.0" --title "Stop DB instance" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StopRDSInstance\",\"Region\": \"us-east-1\",\"Parameters\":
{\"InstanceId\":[\"rds-instance\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named StopRdsDbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2r2bffv9u6q4m" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopRdsDbParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
"DocumentName": "AWSManagedServices-StopRDSInstance",
"Region": "us-east-1",
"Parameters": {
  "InstanceId": [ "rds-instance" ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it StopRdsDbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StopRdsDbRfc.json
```

4. Modify and save the StopRdsDbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2r2bffv9u6q4m",
"Title":                "RDS-STOP-DB-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the StopRdsDbRfc file:

```
aws amscm create-rfc --cli-input-json file://StopRdsDbRfc.json --execution-
parameters file://StopRdsDbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This command doesn't apply to Aurora MySQL and Aurora PostgreSQL. For Aurora clusters, use a Management | Other | Other |Update (ct-0xdawir96cy7k) change type and request StopDBCluster instead.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2r2bffv9u6q4m.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

RDS Database Stack | Update

Modify the properties of an Amazon Relational Database Service (RDS) DB instance created using ct-2z60dyvto9g6c, version 3.0.

Full classification: Management | Advanced stack components | RDS database stack | Update

Change Type Details

Change type ID	ct-12w49boaiwtzp
Current version	3.0
Expected execution duration	360 minutes
AWS approval	Required

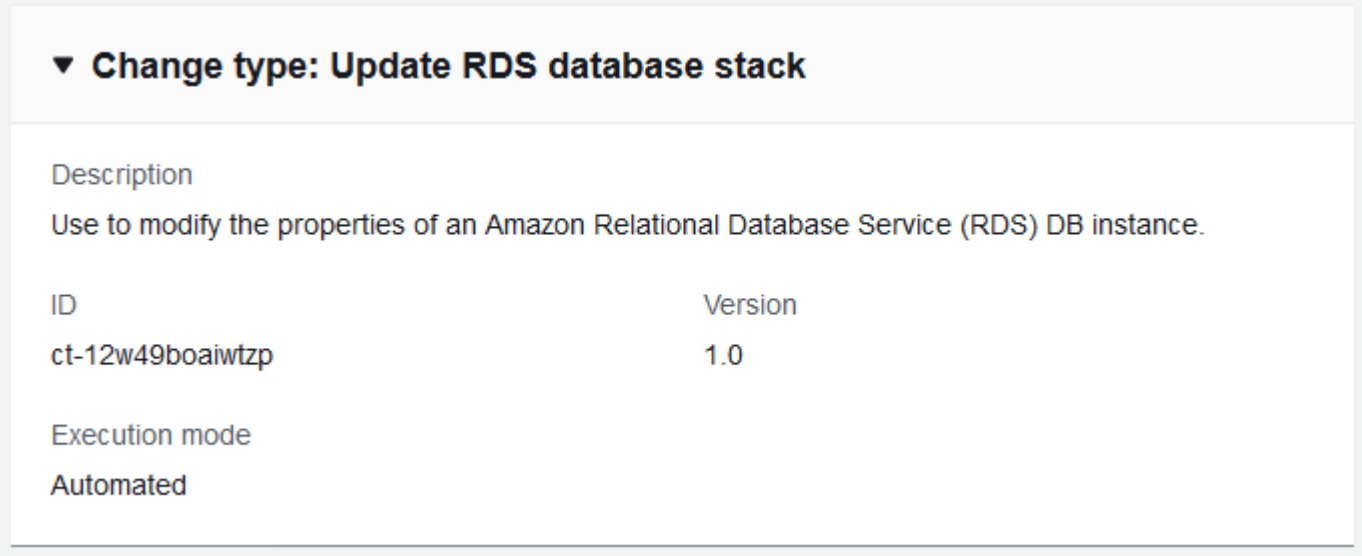
Customer approval	Not required
Execution mode	Automated

Additional Information

Update DB stack

Updating an RDS Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-12w49boaiwtzp" --change-type-version "1.0"
 --title "RDS_UPDATE" --execution-parameters "{\"VpcId\":\"VPC_ID\",\"StackId\":
\"STACK_ID\",\"Parameters\":{\"RDSBackups\":true,\"RDSInstanceType\":\"db.m3.medium\",
\"RDSIOPS\":0,\"RDSMultiAZ\":true,\"RDSPreferredBackupWindow\":\"22:00-23:00\",
\"RDSPreferredMaintenanceWindow\":\"wed:03:32-wed:04:02\",\"RDSStorageType\":\"gp2\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateRdsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-12w49boaiwtzp" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateRdsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":          "Update-RDS-DB",
   "VpcId":                "VPC_ID",
   "StackId":              "STACK_ID",
   "Parameters":   {
       "RDSAllocatedStorage":      80,
       }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateRdsRfc.json
```

4. Modify and save the UpdateRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-12w49boaiwtzp",
"Title":                "RDS-Update-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateRdsRfc.json --execution-
parameters file://UpdateRdsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update an RDS stack for Aurora, see Update DB (for Aurora).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-12w49boaiwtzp.

Example: Required Parameters

```
{
  "VpcId": "vpc-01234567890abcdef",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "RDSAllocatedStorage": 50
  }
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-01234567",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "RDSAllocatedStorage": 50,
    "RDSAllowMajorVersionUpgrade": true,
    "RDSAutoMinorVersionUpgrade": true,
    "RDSBackupRetentionPeriod": 7,
    "RDSDBParameterGroupName": "default.sqlserver-ex-13.0",
    "RDSDeletionProtection": true,
    "RDSDomain": "d-1234567890",
    "RDSDomainIAMRoleName": "customer_amazon_rds_directory_service_access_role",
    "RDSBackups": true,
    "RDSEngineVersion": "5.6.27",
    "RDSInstanceType": "db.m3.medium",
    "RDSIOPS": 0,
    "RDSMasterUserPassword": "$tr0n9PA55w0Rd",
    "RDSMultiAZ": false,
    "RDSOptionGroupName": "default:sqlserver-ex-13-00",
    "RDSPerformanceInsights" : "true",
    "RDSPerformanceInsightsKMSKey":"arn:aws:kms:us-east-1:123456789012:key/2590cd3a-
f979-49db-adec-d213775385af",
    "RDSPerformanceInsightsRetentionPeriod":"7",
    "RDSPreferredBackupWindow": "22:00-23:00",
    "RDSPreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "RDSStorageType": "gp2"
```

```
  }
}
```

RDS Database Stack | Update (For Aurora)

Modify the properties of an existing AWS Relational Database Service (RDS) Aurora stack created using CT ID ct-2jvzjwunghrhy, version 1.0.

Full classification: Management | Advanced stack components | RDS database stack | Update (for Aurora)

Change Type Details

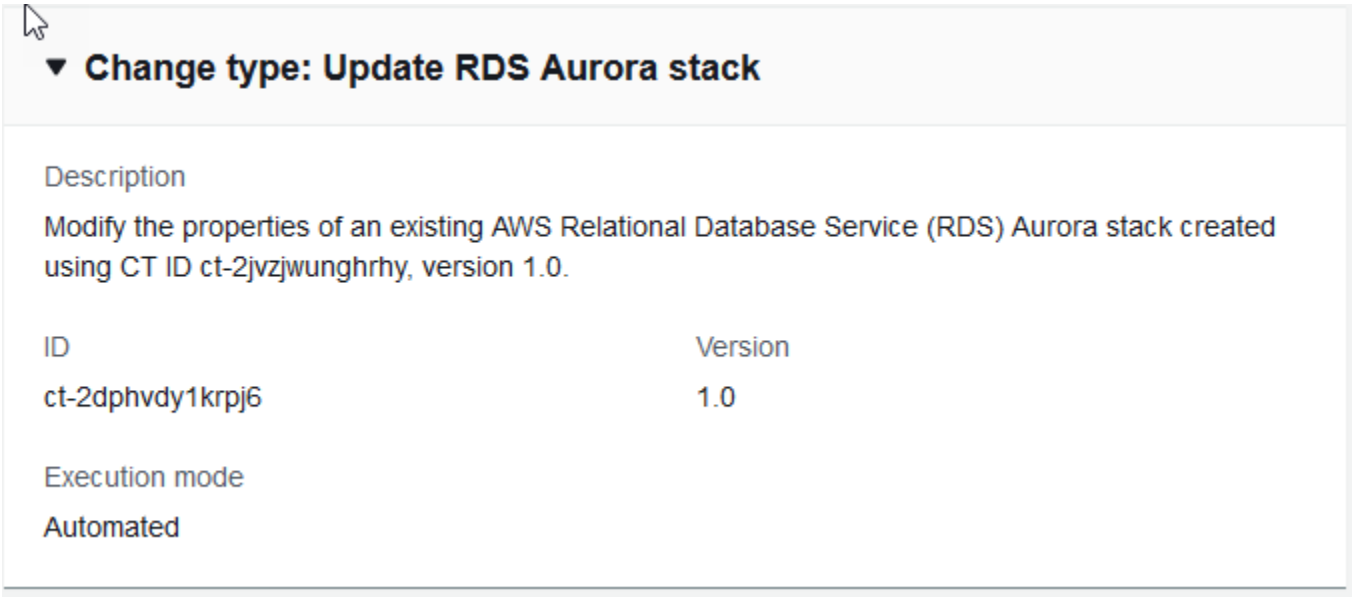
Change type ID	ct-2dphvdy1krpj6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update DB (for Aurora)

Updating an RDS Aurora Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS Aurora Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --region us-east-1 --change-type-id "ct-2dphvdy1krpj6" --change-
type-version "1.0" --title "Test - Update Aurora RDS"  --execution-parameters "{\"VpcId
\":\"VPC_ID\",\"StackId\":\"STACK_ID\",\"Parameters\":{\"AutoMinorVersionUpgrade
\":\"true\",\"BackupRetentionPeriod\":5,\"EngineVersion\":\"10.4\",\"InstanceType
\":\"db.r4.large\",\"MultiAZ\":\"true\",\"PerformanceInsights\":\"true\",
\"PerformanceInsightsKMSKey\":\"default\",\"PerformanceInsightsRetentionPeriod
\":\"7\",\"Port\":\"1151\",\"PreferredBackupWindow\":\"22:00-23:00\",
\"PreferredMaintenanceWindow\":\"wed:03:32-wed:04:02\",\"MasterUserPassword\":\"PW\"}}"
```

TEMPLATE CREATE (all parameters shown):

1. Output the execution parameters for this change type to a JSON file named UpdateAuroraRdsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2dphvdy1krpj6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateAuroraRdsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "VpcId": "VPC_ID",
     "StackId": "STACK_ID",
     "Parameters": {
       "AutoMinorVersionUpgrade": "true",
       "BackupRetentionPeriod": 5,
       "EngineVersion": "10.4",
       "InstanceType": "db.r4.large",
       "MultiAZ": "true",
       "PerformanceInsights": "true",
       "PerformanceInsightsKMSKey": "default",
       "PerformanceInsightsRetentionPeriod": "7",
       "Port": "1151",
       "PreferredBackupWindow": "22:00-23:00",
       "PreferredMaintenanceWindow": "wed:03:32-wed:04:02",
       "MasterUserPassword": "**********"
     }
   ```

```
}
```

3. Output the JSON template to a file in your current folder; this example names it UpdateAuroraRdsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateAuroraRdsRfc.json
```

4. Modify and save the UpdateAuroraRdsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2dphvdy1krpj6",
"Title":                "RDS-Aurora-Update-RFC"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateAuroraRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateAuroraRdsRfc.json --execution-
parameters file://UpdateAuroraRdsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update a non-Aurora RDS stack, see Update DB stack.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2dphvdy1krpj6.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678901234567",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "AutoMinorVersionUpgrade": "true",
    "BackupRetentionPeriod": 7,
    "InstanceType": "db.serverless",
    "MasterUserPassword": "dbpassword",
    "MultiAZ": "true",
    "PerformanceInsights": "true",
    "PerformanceInsightsKMSKey": "default",
    "PerformanceInsightsRetentionPeriod": "7",
    "Port": "1150",
    "PreferredBackupWindow": "22:00-23:00",
    "PreferredMaintenanceWindow": "wed:03:32-wed:04:02",
    "ServerlessScalingMinCapacity": 1.0,
    "ServerlessScalingMaxCapacity": 2.0
  }
}
```

RDS Database Stack | Update Deletion Protection

Update the DeletionProtection setting for the specified RDS instance or cluster. The RDS instance or cluster can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS was provisioned through CFN ingestion.

Full classification: Management | Advanced stack components | RDS database stack | Update deletion protection

Change Type Details

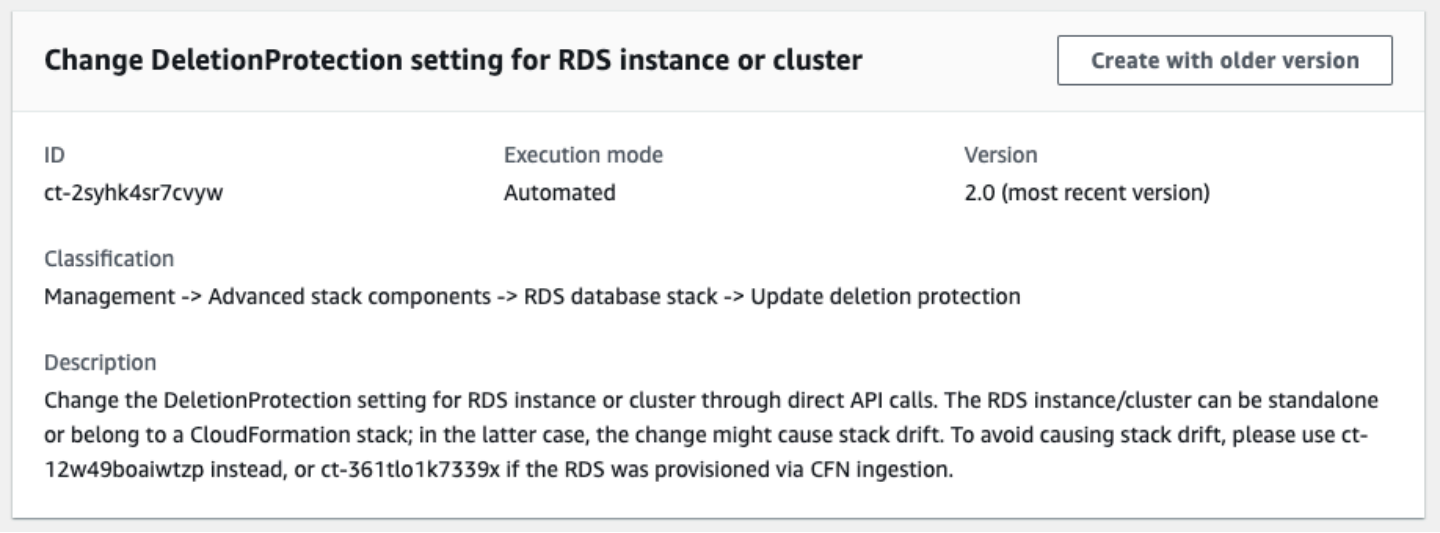
Change type ID	ct-2syhk4sr7cvyw
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update DB deletion protection

Updating an RDS stack deletion protection with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS stack deletion protection with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-2syhk4sr7cvyw" \
--change-type-version "2.0" --title "Update RDS deletion protection" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
UpdateRDSDeletionProtection\",\"Region\":\"us-east-1\",\"Parameters\":
{\"DBIdentifierArn\":[\"arn:aws:rds:ap-southeast-2:012345678901:db:myrds\"],
\"DeletionProtection\":[true]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateRdsDeletionProtectionParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2syhk4sr7cvyw"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateRdsDeletionProtectionParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-UpdateRDSDeletionProtection",
  "Region": "ap-southeast-2",
  "Parameters": {
    "DBIdentifierArn": [
      "arn:aws:rds:ap-southeast-2:012345678901:db:myrds"
    ],
    "DeletionProtection": [
      true
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it UpdateRdsDeletionProtectionRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateRdsDeletionProtectionRfc.json
```

4. Modify and save the UpdateRdsDeletionProtectionRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-2syhk4sr7cvyw",
  "ChangeTypeVersion": "2.0",
  "Title": "Update RDS deletion protection"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsDeletionProtectionRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateRdsDeletionProtectionRfc.json --
execution-parameters file://UpdateRdsDeletionProtectionParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. The execution output indicates the action taken: enabled or disabled.

Tips

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2syhk4sr7cvyw.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-UpdateRDSDeletionProtection",
  "Region" : "us-east-1",
  "Parameters" : {
    "DBIdentifierArn" : "arn:aws:rds:us-east-1:123456789012:db:testdbinstance",
    "DeletionProtection" : true
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-UpdateRDSDeletionProtection",
  "Region" : "us-east-1",
  "Parameters" : {
    "DBIdentifierArn" : "arn:aws:rds:us-east-1:123456789012:db:testdbinstance",
    "DeletionProtection" : true
  }
}
```

RDS Database Stack | Update Enhanced Monitoring

Update the Enhanced Monitoring property of an Amazon Relational Database Service (RDS) database instance or cluster. Enhanced Monitoring allows you to collect vital operating system metrics and process information, at the defined granularity.

Full classification: Management | Advanced stack components | RDS database stack | Update enhanced monitoring

Change Type Details

Change type ID	ct-3jx80fquylzhf

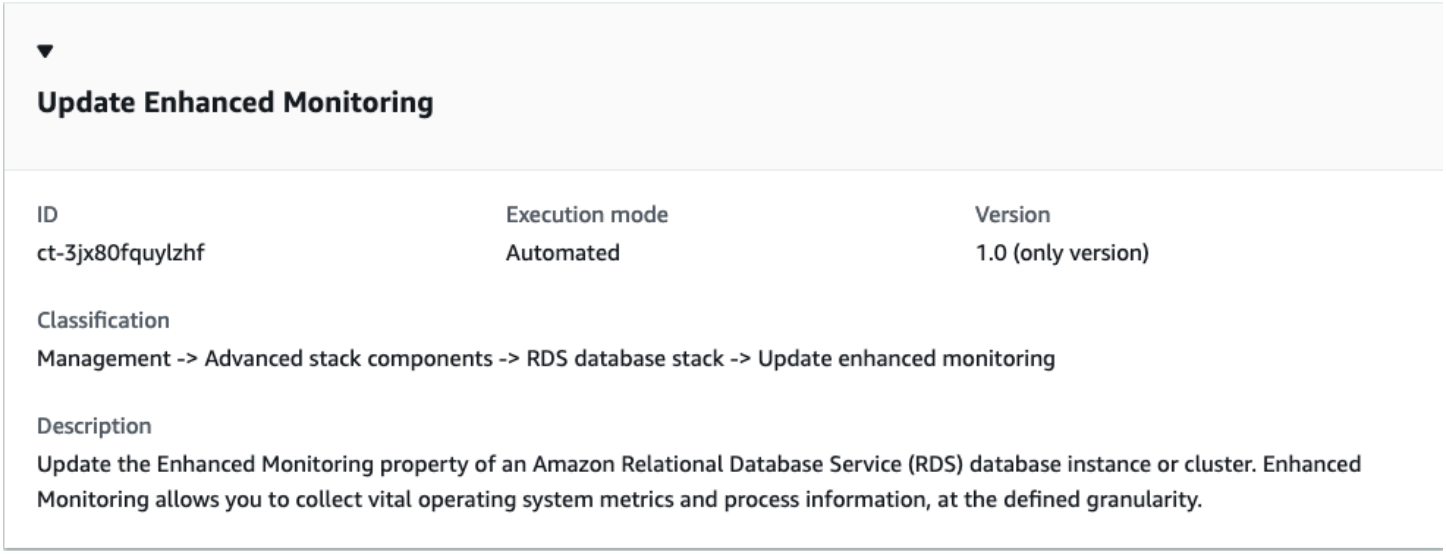
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update Enhanced Monitoring

Updating enhanced monitoring with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating enhanced monitoring with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3jx80fquylzhf" --change-type-
version "1.0" --title "Update Enhanced Monitoring" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-UpdateRDSEnhancedMonitoring\",\"Region
\":\"us-east-1\",\"Parameters\": {\"DBIdentifierArn\":[\"arn:aws:rds:us-
east-1:000000000000:db:testdbinstance\"], \"MonitoringInterval\": [\"60\"],,
\"MonitoringRoleName\": \"ds-monitoring-role\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named RotateRdsCertParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3jx80fquylzhf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateRDSEnhancedMonitoringParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-UpdateRDSEnhancedMonitoring",
       "Region": "us-east-1",
       "Parameters": {
           "DBIdentifierArn": "arn:aws:rds:us-east-1:000000000000:db:testdbinstance",
   ```

```
        "MonitoringInterval": "60",
        "MonitoringRoleName": [
            "rds-monitoring-role"
        ]
    }
}
```

3. Output the JSON template to a file in your current folder; this example names it UpdateRDSEnhancedMonitoringRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateRDSEnhancedMonitoringRfc.json
```

4. Modify and save the UpdateRDSEnhancedMonitoringRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3jx80fquylzhf",
"Title":                "Update Enhanced Monitoring"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRDSEnhancedMonitoringRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateRDSEnhancedMonitoringRfc.json --
execution-parameters file://UpdateRDSEnhancedMonitoringParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3jx80fquylzhf.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateRDSEnhancedMonitoring",
  "Region": "us-east-1",
  "Parameters": {
```

```
    "DBIdentifierArn" : "arn:aws:rds:us-east-1:000000000000:db:testdbinstance",
    "MonitoringInterval" : "60"
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-UpdateRDSEnhancedMonitoring",
  "Region" : "us-east-1",
  "Parameters" : {
    "DBIdentifierArn" : "arn:aws:rds:us-east-1:000000000000:db:testdbinstance",
    "MonitoringInterval" : "60",
    "MonitoringRoleName": "rds-monitoring-role"
  }
}
```

RDS Database Stack | Update Instance Type

Change the DB instance type through direct API calls. The RDS instance can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS instance was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | RDS database stack | Update instance type

Change Type Details

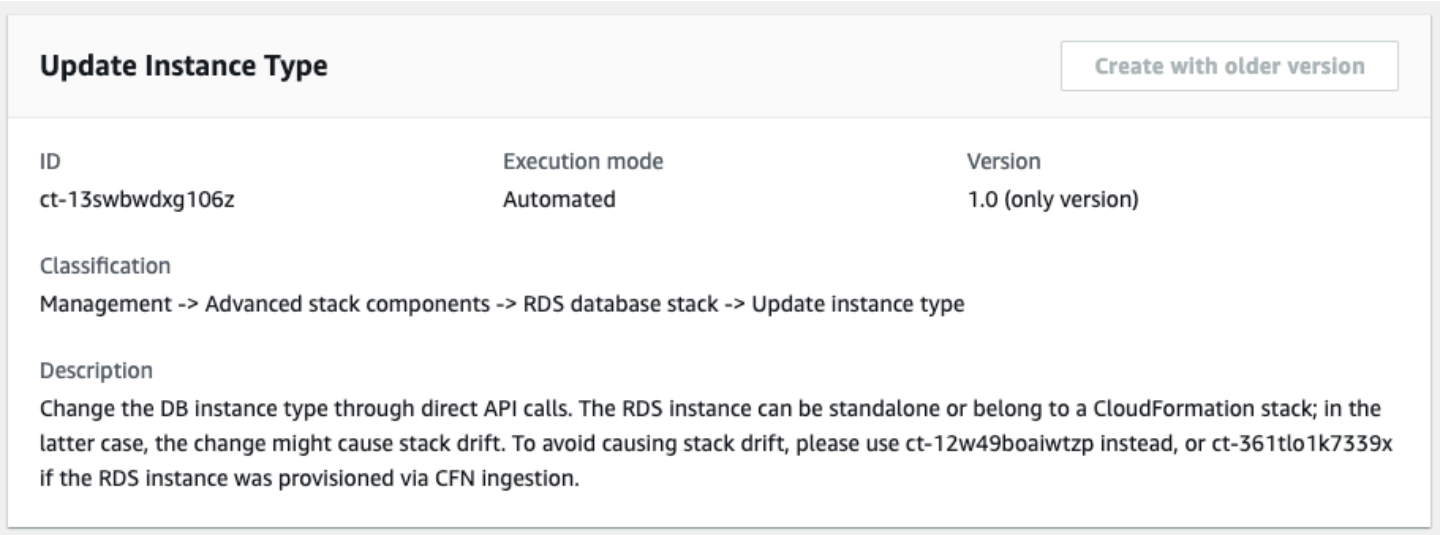
Change type ID	ct-13swbwdxg106z
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update an RDS instance type

Updating an RDS instance type with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS instance type with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-13swbwdxg106z" --change-type-version
 "1.0" --title "Update rds instance type" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateRDSInstanceType\",\"Region\":\"us-east-1\",\"Parameters\":
 {\"DBInstanceIdentifier\":[\"rt123456789\"], \"DBInstanceClass\": [\"db.m4.large\"],
\"ApplyImmediately\": \"true\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateInstanceTypeParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-13swbwdxg106z"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateInstanceTypeParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateRDSInstanceType",
     "Region": "us-east-1",
     "Parameters": {
       "DBInstanceIdentifier": [
         "rt123456789"
       ],
       "DBInstanceClass": [
         "db.m4.large"
       ],
       "ApplyImmediately": "false"
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateInstanceTypeRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateInstanceTypeRfc.json
```

4. Modify and save the UpdateInstanceTypeRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-13swbwdxg106z",
"Title":                "Update RDS instance type"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateInstanceTypeRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateInstanceTypeRfc.json --
execution-parameters file://UpdateInstanceTypeParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update an RDS stack for Aurora, see RDS Database Stack | Update.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-13swbwdxg106z.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateRDSInstanceType",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": [
      "rt123456789"
    ],
    "DBInstanceClass": [
      "db.m4.large"
    ],
    "ApplyImmediately": "false"
  }
}
```

RDS Database Stack | Update Maintenance Window (Review Required)

Update an existing RDS maintenance window, which is a weekly time range (in UTC) during which system maintenance can occur. Changing an RDS maintenance window doesn't result in an outage. If moving this window to the current time, there must be at least 30 minutes between the current time and the end of the current window to ensure pending changes are applied.

Full classification: Management | Advanced stack components | RDS database stack | Update maintenance window (review required)

Change Type Details

Change type ID	ct-27jjy5wnrfef2
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required

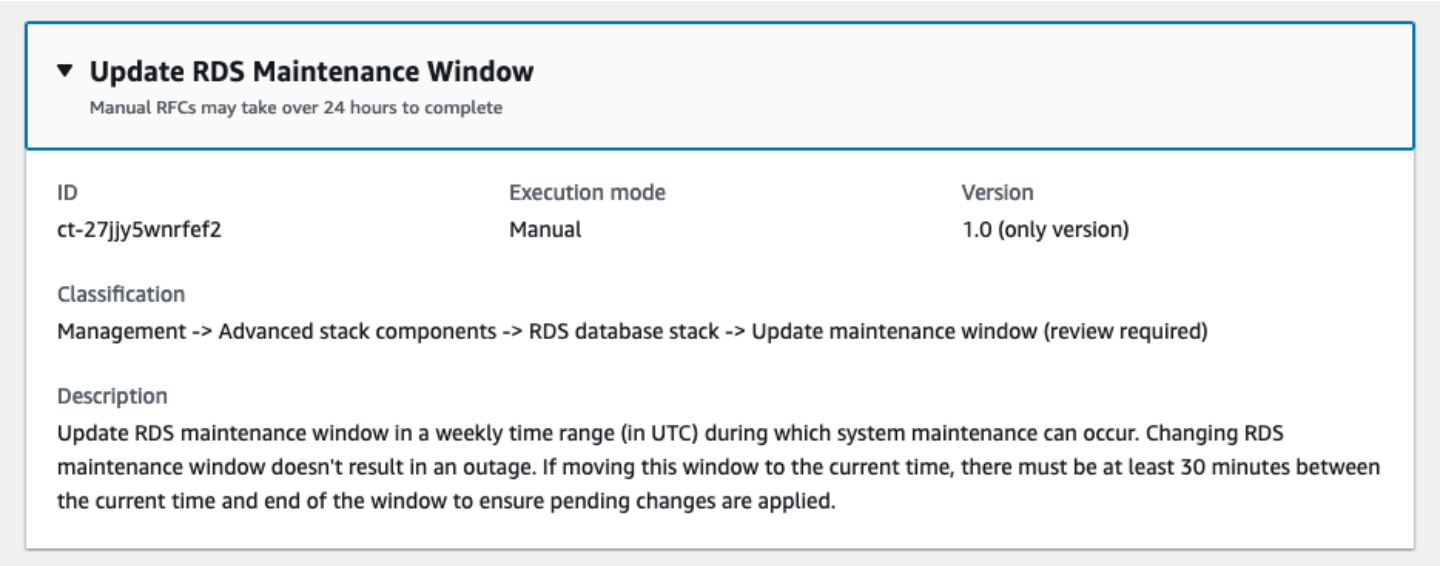
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update RDS maintainance window (review required)

Updating an RDS maintainance window with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS maintainance window with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-27jjy5wnrfef2" --change-type-version "1.0"
 --title "Update RDS Maintenance Window" --execution-parameters "{\"DBIdentifierArn\":
 \"arn:aws:rds:us-east-1:123456789101:db:database-1\", \"PreferredMaintenanceWindow\":
 \"Sun:04:00-Sun:04:30\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateRDSMaintenanceWindowParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-27jjy5wnrfef2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateRDSMaintenanceWindowParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789101:db:database-1",
     "PreferredMaintenanceWindow": "Sun:04:00-Sun:04:30"
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateRDSMaintenanceWindowRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateRDSMaintenanceWindowRfc.json
   ```

4. Modify and save the UpdateRDSMaintenanceWindowRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-27jjy5wnrfef2",
  "Title": "Update RDS Maintenance Window"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRDSMaintenanceWindowRfc.json file:

```
aws amscm create-rfc --cli-input-json file://UpdateRDSMaintenanceWindowRfc.json --
execution-parameters file://UpdateRDSMaintenanceWindowParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-27jjy5wnrfef2.

Example: Required Parameters

```
{
  "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
  "PreferredMaintenanceWindow": "Sun:05:00-Sun:05:30"
}
```

Example: All Parameters

```
{
  "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
  "PreferredMaintenanceWindow": "Sun:05:00-Sun:05:30"
}
```

RDS Database Stack | Update Master User Password

Update the MasterUserPassword property of an Amazon Relational Database Service (RDS) database instance.

Full classification: Management | Advanced stack components | RDS database stack | Update master user password

Change Type Details

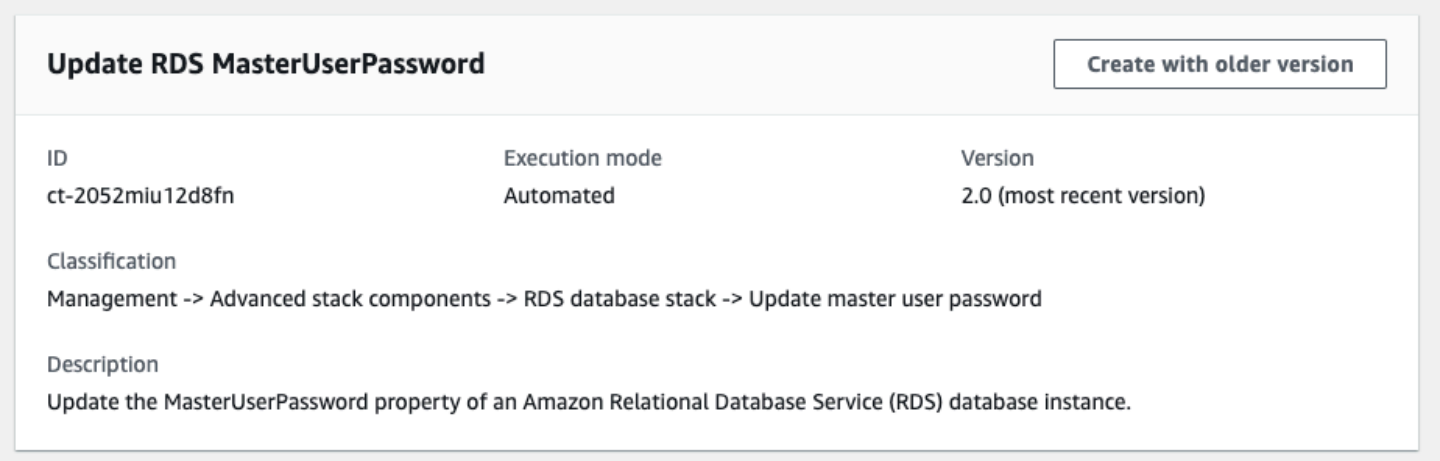
Change type ID	ct-2052miu12d8fn
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update DB master user password

Update the RDS master user password with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Update an RDS master user password with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2052miu12d8fn" --change-type-version "2.0"
 --title "Update RDS master user password" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateInstanceMasterUserPasswordV2\",\"Region\":\"us-east-1\",
\"Parameters\": {\"DBInstanceIdentifier\":[\"myrdsinstance\"], \"SecretName\":
 \"my_secret_name\", \"SecretKey\": \"my_secret_key\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateRdsMPParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2052miu12d8fn" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateRdsMPParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateInstanceMasterUserPasswordV2",
     "Region": "ap-southeast-2",
     "Parameters": {
       "DBInstanceIdentifier": "myrdsinstance",
       "SecretName": "my_secret_name",
       "SecretKey": "my_secret_key",
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateRdsMPRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateRdsMPRfc.json
   ```

4. Modify and save the UpdateRdsMPRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2052miu12d8fn",
"Title":                ""Update RDS master user password"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsMPRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateRdsMPRfc.json --execution-
parameters file://UpdateRdsMPParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the RDS, look in the execution output: Use the "stack_id" to view the RDS in the Cloud Formation Console.

7. You are now able to manage the database via a database management tool such as SQL server management studio. You do not have to request access from AMS.

Tips

- Before you use this CT, you must first store your new master user password in one of the following:
  - AWS Systems Manager (SSM) Parameter Store
  - AWS Secrets Manager
- To use **SSM (AWS Systems Manager) Parameter Store**

  SSM Parameter Store

CT Parameter	Value
SSMParameter	"*my_ssm_parameter_name* "

- To use **AWS Secrets Manager**

AWS Secrets Manager

CT Parameter	Value
SecretName	"*my_secret_name* "
SecretKey	"*my_secret_key* "

- To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.
- For an informal comparison of Secrets Manager and SSM Parameter Store, see AWS — Difference between Secrets Manager and Parameter Store (Systems Manager).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2052miu12d8fn.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateInstanceMasterUserPasswordV2",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": "rt123456789",
    "SecretName": "mysecret",
    "SecretKey": "mypassword"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateInstanceMasterUserPasswordV2",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": "rt123456789",
    "SecretName": "mysecret",
    "SecretKey": "mypassword"
  }
```

```
}
```

RDS Database Stack | Update MultiAZ Setting

Change the DB instance MultiAZ value through direct API calls. The MultiAZ setting determines whether or not the DB instance is deployed across multiple availability zones (AZs). The RDS instance can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS instance was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | RDS database stack | Update MultiAZ setting

Change Type Details

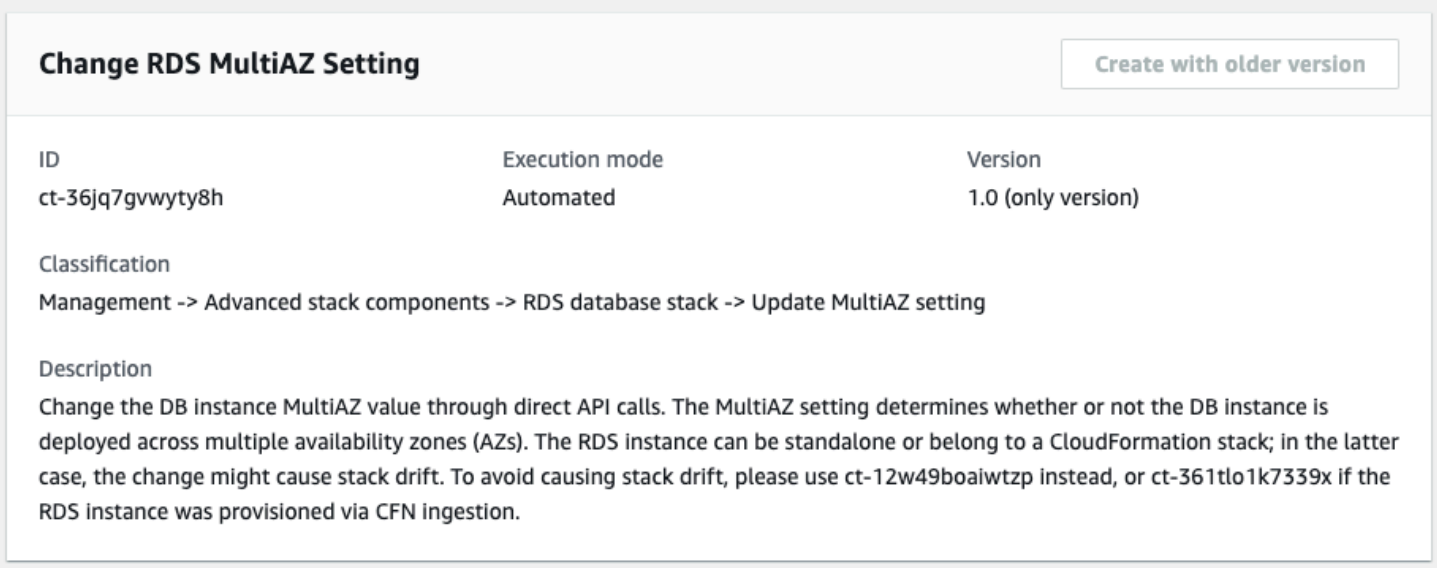
Change type ID	ct-36jq7gvwyty8h
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update an RDS multi-AZ deployment

Updating an RDS multi-AZ deployment with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an RDS multi-AZ deployment with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-36jq7gvwyty8h" --change-type-version
 "1.0" --title "Update RDS Multiple AZ" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateRDSMultiAZ\",\"Region\":\"us-east-1\",\"Parameters\":
 {\"DBInstanceIdentifier\":[\"rt123456789\"], \"MultiAZ\": \"true\",\"ApplyImmediately
\": \"true\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateMultipleAzParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-36jq7gvwyty8h"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateMultipleAzParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateRDSMultiAZ",
     "Region": "us-east-1",
     "Parameters": {
       "DBInstanceIdentifier": [
         "rt123456789"
       ],
       "MultiAZ": "true",
       "ApplyImmediately": "false"
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateMultipleAzRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateMultipleAzRfc.json
   ```

4. Modify and save the UpdateMultipleAzRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-36jq7gvwyty8h",
"Title":                "Update RDS Multiple AZ"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateMultipleAzRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateMultipleAzRfc.json --execution-
parameters file://UpdateMultipleAzParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update an RDS stack for Aurora, see RDS Database Stack | Update.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-36jq7gvwyty8h.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateRDSMultiAZ",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": [
      "rt123456789"
    ],
    "MultiAZ": "true",
    "ApplyImmediately": "false"
  }
}
```

RDS Database Stack | Update Performance Insights (Review Required)

Update Performance Insights for a DB instance or Multi-AZ DB cluster. Amazon RDS Performance Insights is a database performance tuning and monitoring feature that helps you assess the load on your database. You can change settings, enable, or disable the feature.

Full classification: Management | Advanced stack components | RDS database stack | Update Performance Insights (review required)

Change Type Details

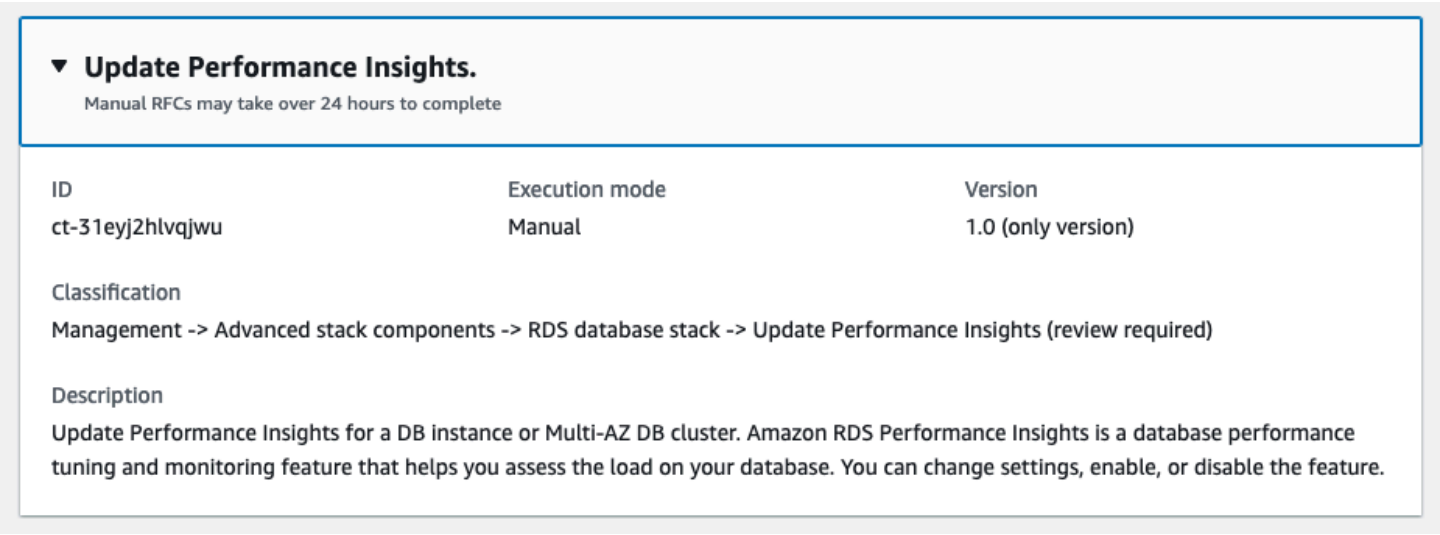
Change type ID	ct-31eyj2hlvqjwu
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Manual

Additional Information

Update RDS performance insights (review required)

Updating an RDS performance insights with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating performance insights with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-31eyj2hlvqjwu" --change-type-
version "1.0" --title "Update Performance Insights." --execution-parameters
 "{\"DBIdentifierArn\": \"arn:aws:rds:us-east-1:123456789012:cluster:database-1\",
\"PerformanceInsights\": \"true\", \"PerformanceInsightsKMSKeyId\": \"default\",
 \"PerformanceInsightsRetentionPeriod\": \"7 days\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdatePerformanceInsightsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-31eyj2hlvqjwu"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdatePerformanceInsightsParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789101:cluster:database-1",
     "PerformanceInsights": "true",
     "PerformanceInsightsKMSKeyId": "default",
     "PerformanceInsightsRetentionPeriod": "7 days"
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdatePerformanceInsightsRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdatePerformanceInsightsRfc.json
   ```

4. Modify and save the UpdatePerformanceInsightsRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   ```

```
"ChangeTypeId":          "ct-31eyj2hlvqjwu",
"Title":                 "Update Performance Insights"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdatePerformanceInsightsRfc.json --
execution-parameters file://UpdatePerformanceInsightsParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update an RDS stack for Aurora, see RDS Database Stack | Update.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-31eyj2hlvqjwu.

Example: Required Parameters

```
{
  "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
  "PerformanceInsights": "true"
}
```

Example: All Parameters

```
{
  "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
  "PerformanceInsights": "true",
  "PerformanceInsightsKMSKeyId": "arn:aws:kms:us-
east-1:123456789012:key/58c399bf-1662-4d55-8bbe-fb6d26bd72b9",
  "PerformanceInsightsRetentionPeriod": "7 days"
}
```

RDS Database Stack | Update Storage

Change the RDS instance storage type, capacity or IOPS through direct API calls. The RDS instance can be standalone or belong to a CloudFormation stack, in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS instance was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | RDS database stack | Update Storage

Change Type Details

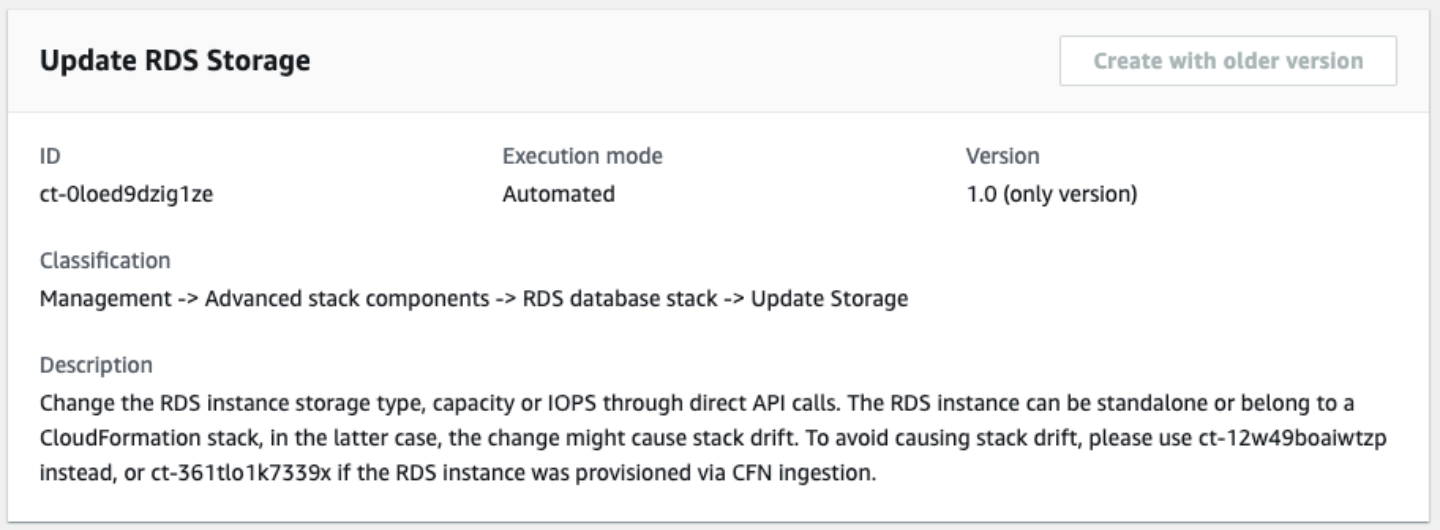
Change type ID	ct-0loed9dzig1ze
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update RDS storage

Updating RDS storage with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating RDS storage with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0loed9dzig1ze" --change-type-version
 "1.0" --title "Update RDS storage" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateRDSStorage\",\"Region\":\"us-east-1\",\"Parameters\":
 {\"DBInstanceIdentifier\":[\"rt123456789\"], \"AllocatedStorage\": [\"100\"],,
\"ApplyImmediately\": \"true\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateStorageParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0loed9dzig1ze" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateStorageParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateRDSStorage",
     "Region": "us-east-1",
     "Parameters": {
       "DBInstanceIdentifier": [
         "rt123456789"
       ],
       "AllocatedStorage": [
         "100"
       ],
       "ApplyImmediately": "false"
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateStorageRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateStorageRfc.json
   ```

4. Modify and save the UpdateStorageRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0loed9dzig1ze",
"Title":                "Update RDS storage"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateStorageRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateStorageRfc.json --execution-
parameters file://UpdateStorageParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> AMS employs drift detection on certain stacks, including RDS stacks, to determine if configuration changes. The AMS disallows updates to an RDS stack that has been determined to have configuration drift. The RFC will fail with the following error message: "Update cannot be performed on this stack, please contact AMS for further assistance."

To learn more about Amazon RDS, including size recommendations, see Amazon Relational Database Service Documentation.

To update an RDS stack for Aurora, see RDS Database Stack | Update.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0loed9dzig1ze.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateRDSStorage",
  "Region": "us-east-1",
  "Parameters": {
    "DBInstanceIdentifier": [
      "rt123456789"
    ],
    "AllocatedStorage": [
      "1000"
    ],
    "StorageType": [
      "gp3"
    ],
    "Iops": [
      "10000"
    ],
    "ApplyImmediately": "true"
  }
}
```

RDS Snapshot | Delete

Delete DB instance or cluster snapshots. This document only supports deletion of 'manual' and 'awsbackup' snapshot types. If the snapshot is being copied, the copy operation is terminated. The snapshot must be in available state to be deleted. If one or more snapshots cannot be deleted, automation fails. Up to 20 snapshots can be deleted in one execution.

Full classification: Management | Advanced stack components | RDS snapshot | Delete

Change Type Details

Change type ID	ct-0idxb0xsg1ui6
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required

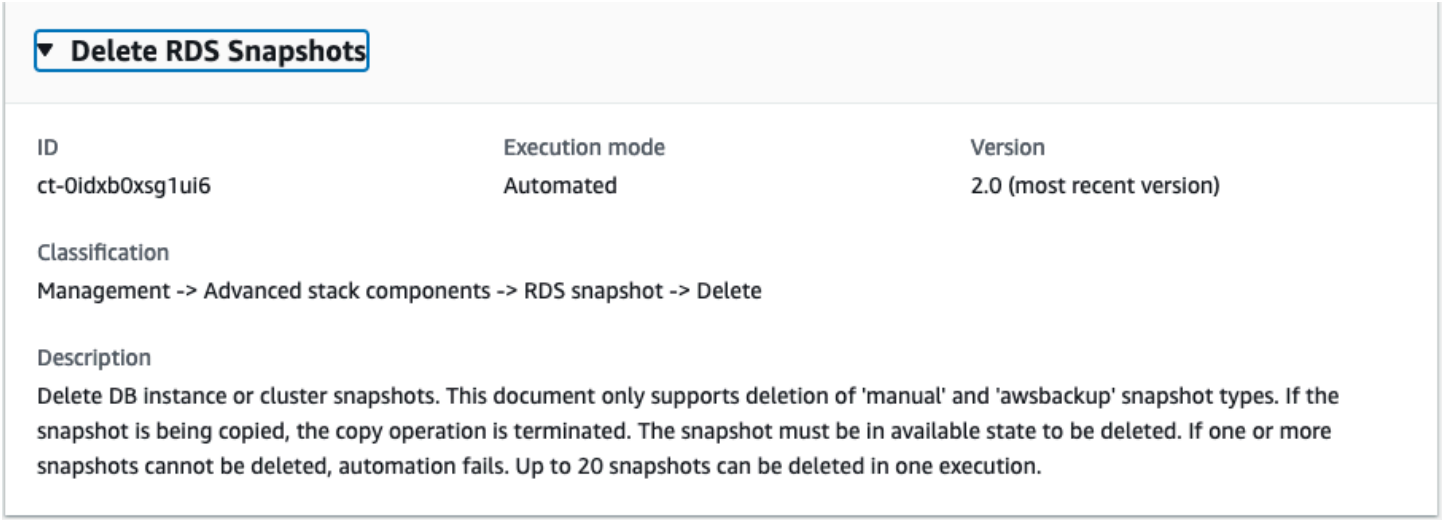
Execution mode	Automated

Additional Information

Delete RDS snapshots

Deleting RDS Snapshots with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting RDS Snapshots with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0idxb0xsg1ui6" --change-type-version
 "2.0" --title "Delete RDS Snapshots" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-DeleteRDSSnapshots\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"SnapshotNamesOrArns\": [\"snapshot1\", \"snapshot2\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named DeleteRdsDbSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0idxb0xsg1ui6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteRDSSnapshotsGroupParameters.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DeleteRDSSnapshots",
     "Region": "us-east-1",
     "SnapshotNamesOrArns": ["snapshot1","snapshot2"]
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it DeleteRdsDbSnapshotRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteRDSSnapshots.json
   ```

4. Modify and save the DeleteRdsDbSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-0idxb0xsg1ui6",
  "Title": "Delete RDS Snapshots"
}
```

5. Create the RFC, specifying the DeleteRDSSnapshots.json file and the execution parameters file, DeleteRDSSnapshotsGroupParameters.json:

```
aws amscm create-rfc --cli-input-json file://DeleteRDSSnapshots.json --execution-
parameters file://DeleteRDSSnapshotsGroupParameters.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about RDS snapshots, see Backing up and restoring an Amazon RDS DB instance.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0idxb0xsg1ui6.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteRDSSnapshotsV2",
  "Region": "us-east-1",
  "Parameters": {
    "SnapshotNamesOrArns": [
      "dbsnapshot",
      "arn:aws:rds:us-east-1:945533541580:snapshot:db2-snapshot",
      "arn:aws:rds:us-east-1:945533541580:cluster-snapshot:db2-snapshot"
```

```
    ]
  }
}
```

RDS Snapshot | Share

Share a snapshot of an Amazon Relational Database Service (RDS) database (DB) instance with another AMS account. Only snapshots encrypted with managed KMS keys can be shared.

Full classification: Management | Advanced stack components | RDS snapshot | Share

Change Type Details

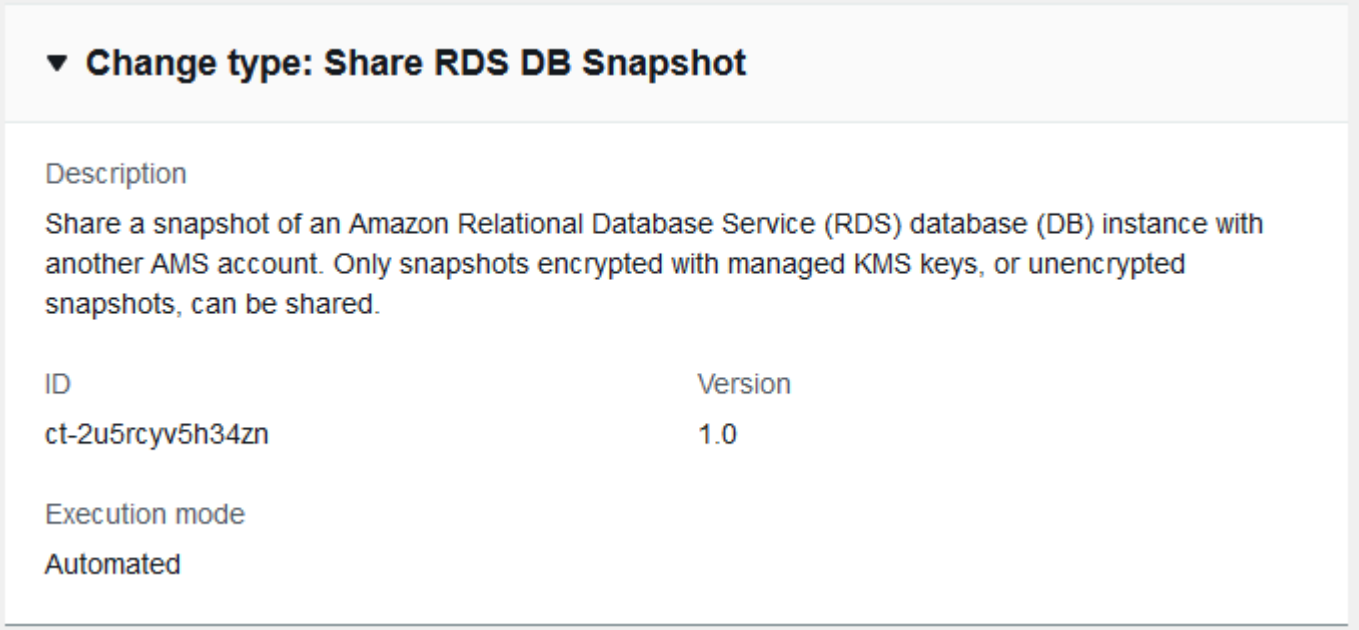
Change type ID	ct-2u5rcyv5h34zn
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Share RDS snapshot

Sharing an RDS DB Snapshot with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Sharing an RDS DB Snapshot with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2u5rcyv5h34zn" --change-type-version
 "1.0" --title "Share DB snapshot" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ShareDBSnapshot\",\"Region\":\"us-east-1\",\"Parameters\":
{\"DBSnapshotName\":[\"rds-db-snapshot\"],\"AccountId\":[\"012345678912\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named ShareRdsDbSnapshotParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2u5rcyv5h34zn"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ShareRdsDbSnapshotParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-ShareDBSnapshot",
     "Region": "us-east-1",
     "Parameters": {
       "DBSnapshotName": [
         "rds-db-snapshot"
       ],
       "AccountId": [
         "012345678912"
       ]
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it ShareRdsDbSnapshotRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ShareRdsDbSnapshotRfc.json
   ```

4. Modify and save the ShareRdsDbSnapshotRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2u5rcyv5h34zn",
  "Title": "Share DB Snapshot"
}
```

5. Create the RFC, specifying the execution parameters file and the ShareRdsDbSnapshotRfc file:

```
aws amscm create-rfc --cli-input-json file://ShareRdsDbSnapshotRfc.json --
execution-parameters file://ShareRdsDbSnapshotParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about RDS, see the RDS User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2u5rcyv5h34zn.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ShareDBSnapshot",
  "Region": "us-east-1",
  "Parameters": {
    "DBSnapshotName": ["dbsnapshot"],
    "AccountId": ["012345678912"]
  }
}
```

Redshift | Pause Cluster

Pause an Amazon Redshift cluster. If a recent snapshot is not available, a temporary manual snapshot is created with a retention period of one day. This snapshot is deleted towards the end of execution for both success and failure scenarios. It is safe for AMS to delete this snapshot as pausing the cluster creates an automated snapshot by default.

Full classification: Management | Advanced stack components | Redshift | Pause cluster

Change Type Details

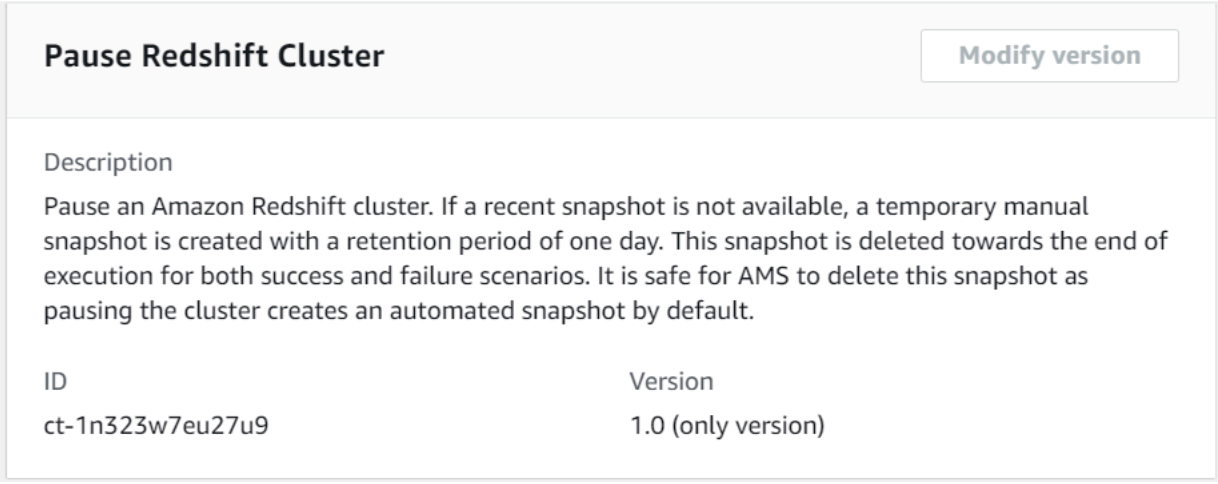
Change type ID	ct-1n323w7eu27u9
Current version	1.0
Expected execution duration	180 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Pause cluster

Pausing a Redshift cluster with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Pausing a Redshift cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1n323w7eu27u9" --change-type-version "1.0"
 --title "Pause Amazon Redshift cluster" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-PauseRedshiftCluster\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ClusterIdentifier\":[\"my-redshift-cluster\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type (ct-1n323w7eu27u9) to a JSON file named PauseRdshftClusterParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-1n323w7eu27u9"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 PauseRdshftClusterParams.json
```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

```
{
  "DocumentName" : "AWSManagedServices-PauseRedshiftCluster",
  "Region" : "us-east-1",
  "Parameters" : {
    "ClusterIdentifier" : [
      "my-redshift-cluster"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it PauseRdshftClusterRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > PauseRdshftClusterRfc.json
```

4. Modify and save the PauseRdshftClusterRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1n323w7eu27u9",
"Title":                "Pause Amazon Redshift cluster"
}
```

5. Create the RFC, specifying the execution parameters file and the PauseRdshftClusterRfc file:

```
aws amscm create-rfc --cli-input-json file://PauseRdshftClusterRfc.json --
execution-parameters file://PauseRdshftClusterParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Redshift, see Amazon Redshift.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1n323w7eu27u9.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-PauseRedshiftCluster",
    "Region": "us-east-1",
    "Parameters": {
      "ClusterIdentifier": ["myredcluster1"]
    }
}
```

Redshift | Resume Cluster

Resume a paused Amazon Redshift cluster.

Full classification: Management | Advanced stack components | Redshift | Resume cluster

Change Type Details

Change type ID	ct-39c5qiasbe4he
Current version	1.0
Expected execution duration	180 minutes
AWS approval	Required
Customer approval	Not required

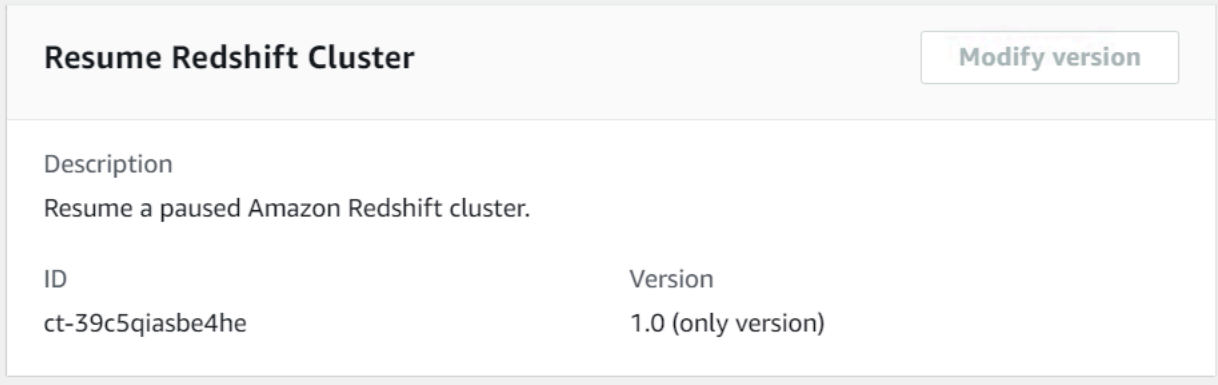
Execution mode	Automated

Additional Information

Resume cluster

Resuming a Redshift cluster with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Resuming a Redshift cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-39c5qiasbe4he" --change-type-version "1.0"
 --title "Resume Amazon Redshift cluster" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ResumeRedshiftCluster\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ClusterIdentifier\":[\"my-redshift-cluster\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type (ct-39c5qiasbe4he) to a JSON file named ResumeRdshftClusterParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-39c5qiasbe4he"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 ResumeRdshftClusterParams.json
```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   Oracle example:

```
{
  "DocumentName" : "AWSManagedServices-ResumeRedshiftCluster",
  "Region" : "us-east-1",
  "Parameters" : {
    "ClusterIdentifier" : [
      "my-redshift-cluster"
    ]
  }
}
```

3. Output the JSON template to a file in your current folder; this example names it ResumeRdshftClusterRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ResumeRdshftClusterRfc.json
```

4. Modify and save the ResumeRdshftClusterRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-39c5qiasbe4he",
"Title":                "Resume Amazon Redshift cluster"
}
```

5. Create the RFC, specifying the execution parameters file and the ResumeRdshftClusterRfc file:

```
aws amscm create-rfc --cli-input-json file://ResumeRdshftClusterRfc.json --
execution-parameters file://ResumeRdshftClusterParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Resume an Amazon Redshift cluster. To learn how to pause an Amazon Redshift cluster, see Pause cluster

To learn more about Amazon Redshift, see Amazon Redshift.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-39c5qiasbe4he.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-ResumeRedshiftCluster",
    "Region": "us-east-1",
    "Parameters": {
      "ClusterIdentifier": ["myredcluster1"]
    }
  }
```

Route 53 Resolver | Associate VPC With Resolver Rule

Associate a VPC with a Route 53 resolver rule, this causes the resolver to forward all DNS queries for the domain name specified in the rule, and that originate in the VPC, to the IP addresses specified in the rule.

Full classification: Management | Advanced stack components | Route 53 Resolver | Associate VPC with resolver rule

Change Type Details

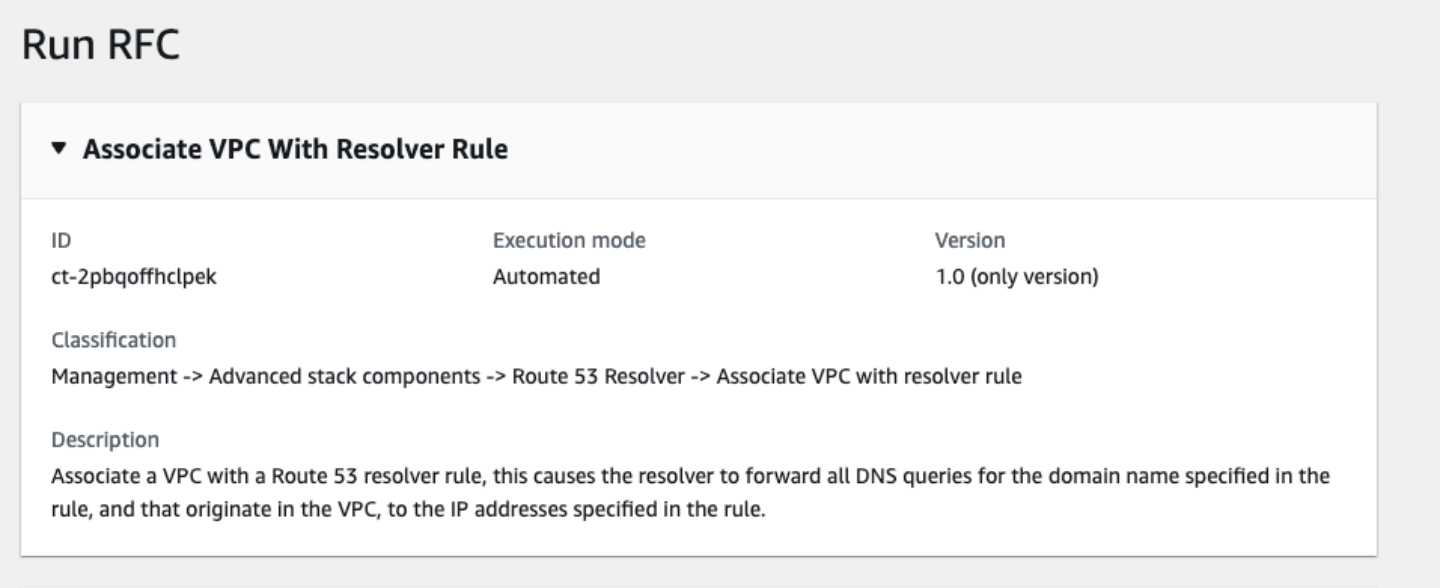
Change type ID	ct-2pbqoffhclpek
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Associate VPC with Resolver Rule

Requesting administrator access with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Requesting administrator access with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title="Associate VPC with Resolver Rule" --ct-
id="ct-2pbqoffhclpek" --ct-version="1.0" --execution-parameters "{\"Description\":
\"Associate VPC with Resolver Rule\",\"ResolverRuleId\":\"rslvr-rr-974b1666869a4d27b\",
\"VPCId\":\"vpc-02a18ed0cd3c17e71\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type; this example names it VPCAssociateResolverRule.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2pbqoffhclpek"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    VPCAssociateResolverRule.json
   ```

2. Modify and save the execution parameters as VPCAssociateResolverRuleParams.json. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-AssociateVPCWithResolverRule",
   ```

```
  "Region": "us-east-1",
  "Parameters": {
    "Name": "resolver-rule-associate-vpc-test",
    "ResolverRuleId": "rslvr-rr-1234567890abcdefg",
    "VPCId": "vpc-1a2b3c4d"
  }
}
```

3. Output the RFC template JSON file; this example names it VPCAssociateResolverRuleRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > VPCAssociateResolverRuleRfc.json
```

4. Modify and save the VPCAssociateResolverRuleRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion" : "1.0",
  "ChangeTypeId" : "ct-2pbqoffhclpek",
  "Title" : "Associate VPC with Resolver Rule "
}
```

5. Create the RFC, specifying the VPCAssociateResolverRuleRfc file and the VPCAssociateResolverRuleParams file:

```
aws amscm create-rfc --cli-input-json file://VPCAssociateResolverRuleRfc.json  --
execution-parameters file:/VPCAssociateResolverRuleParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2pbqoffhclpek.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-AssociateVPCWithResolverRule",
    "Region": "us-east-1",
    "Parameters": {
```

```
      "ResolverRuleId": "rslvr-rr-1234567890abcdefg",
      "VPCId": "vpc-1a2b3c4d"
    }
  }
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-AssociateVPCWithResolverRule",
  "Region": "us-east-1",
  "Parameters": {
    "Name": "resolver-rule-associate-vpc-test",
    "ResolverRuleId": "rslvr-rr-1234567890abcdefg",
    "VPCId": "vpc-1a2b3c4d"
  }
}
```

Route 53 Resolver | Disassociate Resolver Rules from VPC

Removes the associations between specified resolver rules (upto 20) and a specified VPC.

Full classification: Management | Advanced stack components | Route 53 Resolver | Disassociate resolver rules from VPC

Change Type Details

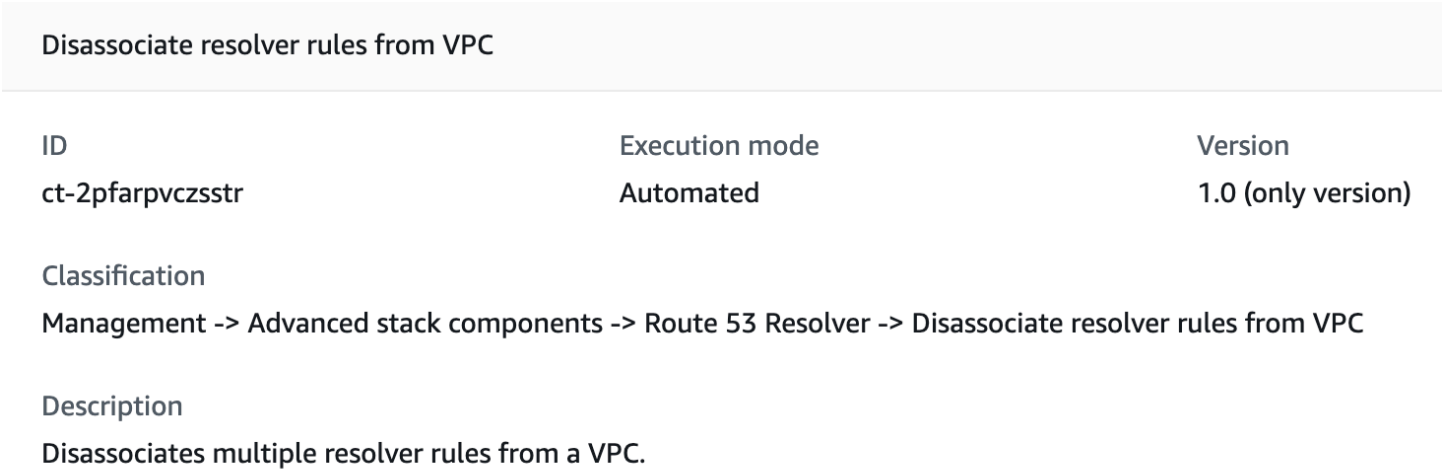
Change type ID	ct-2pfarpvczsstr
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Disassociate resolver rules from VPC

Disassociate resolver rules from a VPC with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Disassociate resolver rules from a VPC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws --profile saml --region us-east-1 amscm create-rfc --change-type-id
 "ct-3e3prksxmdhw8" --change-type-version "2.0" --title "AMI-Create-IC" --
execution-parameters "{\"AMIName\":\"MyAmi\",\"VpcId\":\"VPC_ID\",\"EC2InstanceId\":
\"INSTANCE_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateAmiFromAsgParams.json:

   ```
   aws amscm create-rfc --change-type-id "ct-3e3prksxmdhw8" --change-type-version
    "1.0" --title "Create AMI from an Auto Scaling group" --execution-parameters
    "{\"DocumentName\": \"AWSManagedServices-CreateAmiInAutoScalingGroup\",\"Region
   \": \"us-east-1\",\"Parameters\": {\"AutoScalingGroupName\": [\"stack-ab0123cdef-
   ASG-1ABC2345\"],\"Sysprep\": [\"False\"],\"StopInstance\": [\"False\"]}}"
   ```

2. Modify and save the execution parameters CreateAmiFromAsgParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateAmiInAutoScalingGroup",
     "Region": "us-east-1",
     "Parameters": {
       "AutoScalingGroupName": [
         "stack-ab0123cdef-ASG-1ABC2345"
       ],
       "Sysprep": [
         "False"
       ],
       "StopInstance": [
         "False"
       ]
     }
   }
   ```

3. Output the RFC template JSON file to a file in your current folder; this example names it CreateAmiFromAsgRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateAmiFromAsgRfc.json
```

4. Modify and save the CreateAmiFromAsgRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3e3prksxmdhw8",
  "Title": "Create AMI from an Auto Scaling group"
}
```

5. Create the RFC, specifying the CreateAmiFromAsgRfc file and the CreateAmiFromAsgParams file:

```
aws amscm create-rfc --cli-input-json file://CreateAmiFromAsgRfc.json  --execution-
parameters file://CreateAmiFromAsgParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2pfarpvczsstr.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-DisassociateVPCResolverRules",
    "Region": "us-east-1",
    "Parameters": {
        "ResolverRuleIds": [
            "rslvr-rr-1234567890abcdefg"
        ],
        "VPCId": "vpc-1a2b3c4d"
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-DisassociateVPCResolverRules",
    "Region": "us-east-1",
    "Parameters": {
        "ResolverRuleIds": [
            "rslvr-rr-1234567890abcdefg",
            "rslvr-rr-9876543210abcdefg"
        ],
        "VPCId": "vpc-1a2b3c4d"
    }
}
```

S3 Storage | Add Event Notification

Add an event notification to the specified S3 bucket through direct API calls. The S3 bucket can be standalone or belong to a CloudFormation stack. For buckets in CloudFormation stacks, be aware that stack drift might occur if the bucket was provisioned through CFN ingestion.

Full classification: Management | Advanced stack components | S3 storage | Add event notification

Change Type Details

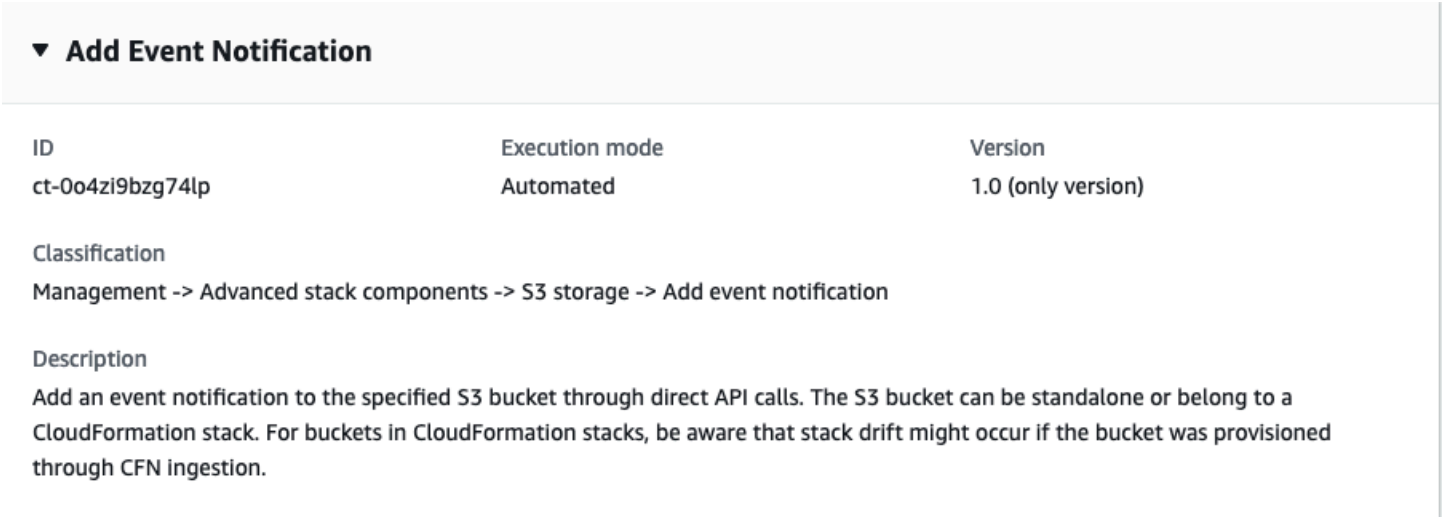
Change type ID	ct-0o4zi9bzg74lp
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add event notification to an Amazon S3 bucket

Add an event notification to an S3 bucket with the Amazon S3 Console

The following is a screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Add an event notification to an S3 bucket with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0o4zi9bzg74lp" --change-type-version
 "1.0" --title "Add event notification" --execution-parameters "{ \"DocumentName
\": \"AWSManagedServices-AddBucketEventNotification\", \"Region\": \"us-
east-1\", \"Parameters\": { \"BucketName\": \"bucketname\", \"EventName\":
 \"eventname\", \"Prefix\": \"foo\", \"Suffix\": \".bar\", \"EventTypes\":
 [ \"s3:ObjectCreated:Post\", \"s3:ObjectCreated:Put\" ], \"DestinationARN\":
 \"arn:aws:lambda:us-east-1:123456789012:function:functionname\" } }"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AddEventNotificationS3Params.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-220bdb8blaixf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AddEventNotificationS3Params.json
   ```

2. Modify and save the AddEventNotificationS3Params file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-AddBucketEventNotification",
     "Region": "us-east-1",
     "Parameters": {
       "BucketName": "bucketname",
       "EventName": "eventname",
       "Prefix": "foo",
       "Suffix": ".bar",
       "EventTypes": [
         "s3:ObjectCreated:Post",
         "s3:ObjectCreated:Put"
       ],
       "DestinationARN": "arn:aws:lambda:us-east-1:123456789012:function:functionname"
     }
   }
   ```

3. Output the RFC template JSON file to a file named AddEventNotificationS3Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AddEventNotificationS3Rfc.json
```

4. Modify and save the AddS3LifecycleConfigRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "1.0",
"ChangeTypeId": "ct-0o4zi9bzg74lp",
"Title": "Add Event Notification"
}
```

5. Create the RFC, specifying the AddEventNotificationS3Rfc file and the AddEventNotificationS3Params file:

```
aws amscm create-rfc --cli-input-json file://AddEventNotificationS3Rfc.json --
execution-parameters file://AddEventNotificationS3Params.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0o4zi9bzg74lp.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-AddBucketEventNotification",
  "Region": "us-east-1",
  "Parameters": {
    "BucketName": "s3-notification-test",
    "EventName": "TestEvent",
    "EventTypes": [
      "s3:ObjectCreated:*",
      "s3:ObjectCreated:Put"
    ],
    "DestinationARN": "arn:aws:lambda:us-east-1:123456789012:function:testfunction"
  }
```

```
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-AddBucketEventNotification",
  "Region": "us-east-1",
  "Parameters": {
    "BucketName": "s3-notification-test",
    "EventName": "TestEvent",
    "EventTypes": [
      "s3:ObjectCreated:*",
      "s3:ObjectCreated:Put",
      "s3:ObjectCreated:Post",
      "s3:ObjectCreated:Copy",
      "s3:ObjectCreated:CompleteMultipartUpload",
      "s3:ObjectRemoved:*",
      "s3:ObjectRemoved:Delete",
      "s3:ObjectRemoved:DeleteMarkerCreated",
      "s3:ObjectRestore:*",
      "s3:ObjectRestore:Post",
      "s3:ObjectRestore:Completed",
      "s3:ObjectRestore:Delete",
      "s3:ReducedRedundancyLostObject",
      "s3:Replication:*",
      "s3:Replication:OperationFailedReplication",
      "s3:Replication:OperationMissedThreshold",
      "s3:Replication:OperationReplicatedAfterThreshold",
      "s3:Replication:OperationNotTracked",
      "s3:LifecycleExpiration:*",
      "s3:LifecycleExpiration:Delete",
      "s3:LifecycleExpiration:DeleteMarkerCreated",
      "s3:LifecycleTransition",
      "s3:IntelligentTiering",
      "s3:ObjectTagging:*",
      "s3:ObjectTagging:Put",
      "s3:ObjectTagging:Delete",
      "s3:ObjectAcl:Put"
    ],
    "DestinationARN": "arn:aws:lambda:us-east-1:123456789012:function:testfunction",
    "Prefix": "testprefix",
    "Suffix": ".jpg"
  }
```

```
}
```

S3 Storage | Add Replication Rule

Add an S3 replication rule to the specified S3 bucket.

Full classification: Management | Advanced stack components | S3 storage | Add replication rule

Change Type Details

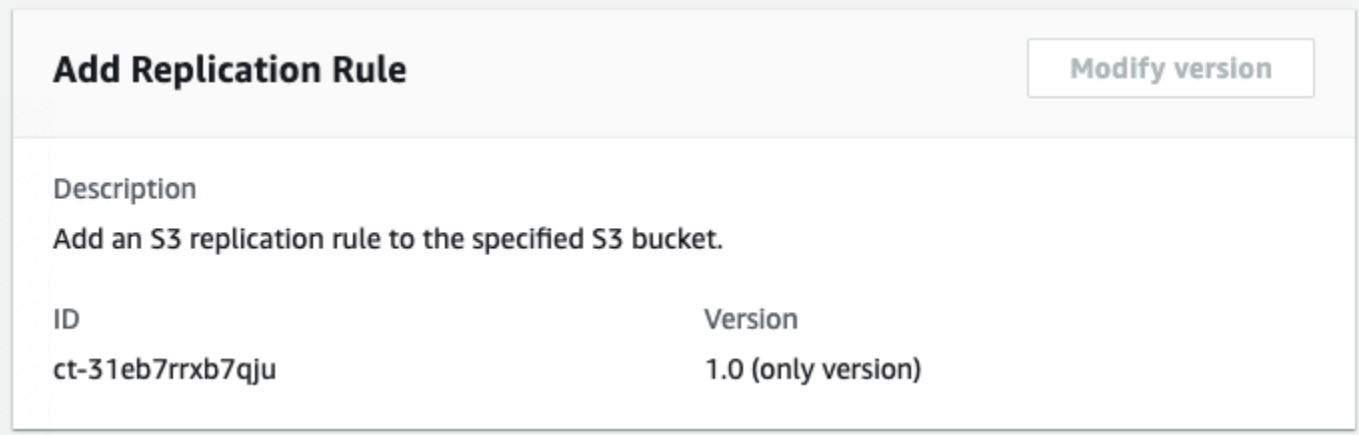
Change type ID	ct-31eb7rrxb7qju
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add replication rule

Adding replication rules to a specified Amazon S3 bucket using the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding replication rules to a specified Amazon S3 bucket using the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

With all parameters for one rule:

```
aws amscm create-rfc --change-type-id "ct-31eb7rrxb7qju" --change-type-version
 "1.0" --title "Put S3 replication rule in the source bucket."--execution-
parameters"{\"DocumentName\":\"AWSManagedServices-PutReplicationRule\",\"Region
\":\"us-east-1\",\"Parameters\":{\"ReplicationRuleName\":[\"test-replication-
all-params\"],\"SourceBucketName\":[\"source-bucket-name\"],\"DestinationAccount
\":[\"123456789012\"],\"DestinationBucketName\":[\"destination-bucket-name\"],
\"ReplicationRole\":[\"arn:aws:iam::123456789012:role/customer_test_s3_replication\"],
\"OwnerTranslation\":[\"false\"],\"DecryptObjectKMSKey\":[\"arn:aws:kms:us-
east-1:123456789012:key/12345678-aaaa-bbbb-cccc-123456789012\"],\"EncryptReplicaKMSKey
\":[\"arn:aws:kms:eu-west-1:012987654321:key/87654321-aaaa-bbbb-cccc-012987654321\"],
\"Prefix\":[\"\"],\"Priority\":[\"1\"]}}"
```

TEMPLATE CREATE:

1. Create and save the PutReplicationRuleParams file.

```
aws amscm get-change-type-version --change-type-id "ct-31eb7rrxb7qju"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 PutReplicationRuleParams.json
```

2.
```
{
    "DocumentName" : "AWSManagedServices-PutReplicationRule",
    "Region": "us-east-1",
    "Parameters": {
      "ReplicationRuleName" : "test-replication-all-params",
      "SourceBucketName" : "source-bucket-name",
      "DestinationAccount" : "123456789012",
      "DestinationBucketName" : "destination-bucket-name",
      "ReplicationRole" : "arn:aws:iam::123456789012:role/
customer_test_s3_replication",
      "OwnerTranslation" : "false",
      "DecryptObjectKMSKey" : ["arn:aws:kms:us-east-1:123456789012:key/12345678-
aaaa-bbbb-cccc-123456789012"],
      "EncryptReplicaKMSKey" : "arn:aws:kms:eu-west-1:012987654321:key/87654321-
aaaa-bbbb-cccc-012987654321",
      "Prefix" : " ",
      "Priority" : "1"
    }
}
```

3. Output the RFC template to a file in your current folder; this example names it PutReplicationRuleRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > PutReplicationRuleRfc.json
```

4. Modify and save the PutReplicationRuleRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeVersion": "1.0",
    "ChangeTypeId": "ct-31eb7rrxb7qju",
    "Title": "Add S3 replication rule in the source bucket."
}
```

5. Create the RFC, specifying the PutReplicationRuleParams file and the PutReplicationRuleRfc file:

```
aws amscm create-rfc --cli-input-json file://PutReplicationRuleRfc.json --
execution-parameters file://PutReplicationRuleParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a new change type that allows you to add replication rules to a specified Amazon S3 bucket. If you want to receive a replication replica in your Amazon S3 bucket, use the S3 storage: Receive replication replica change type.

To learn more about Amazon S3 replication rules , see How do I add a replication rule to an S3 bucket?

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-31eb7rrxb7qju.

Example: Required Parameters

```
{
    "DocumentName" : "AWSManagedServices-PutReplicationRule",
    "Region": "us-east-1",
    "Parameters": {
      "ReplicationRuleName": ["test-replication-only-required-params"],
      "SourceBucketName": ["source-s3-test"],
      "DestinationAccount": ["555555555555"],
      "DestinationBucketName": ["destination-s3-test"],
      "ReplicationRole": ["arn:aws:iam::123456789012:role/
customer_test_s3_replication"]
    }
}
```

Example: All Parameters

```
{
    "DocumentName" : "AWSManagedServices-PutReplicationRule",
    "Region": "us-east-1",
    "Parameters": {
      "ReplicationRuleName": ["test-replication-all-params"],
      "SourceBucketName": ["s3-replication-test"],
      "DestinationAccount": ["555555555555"],
      "DestinationBucketName": ["test-replication-destination"],
      "ReplicationRole": ["arn:aws:iam::123456789012:role/
customer_test_s3_replication"],
      "OwnerTranslation": ["false"],
      "DecryptObjectKMSKey": ["arn:aws:kms:us-east-1:123456789012:key/
bfb30098-2f19-4375-91f5-12345682129a"],
      "EncryptReplicaKMSKey": ["arn:aws:kms:eu-west-1:123456789012:key/
d5e68703-8199-4265-a103-12345637bd47"],
      "Prefix":[""],
      "Priority": ["1"]
    }
}
```

S3 Storage | Delete Policy (Review Required)

Use to delete an S3 bucket policy.

Full classification: Management | Advanced stack components | S3 storage | Delete policy (review required)

Change Type Details

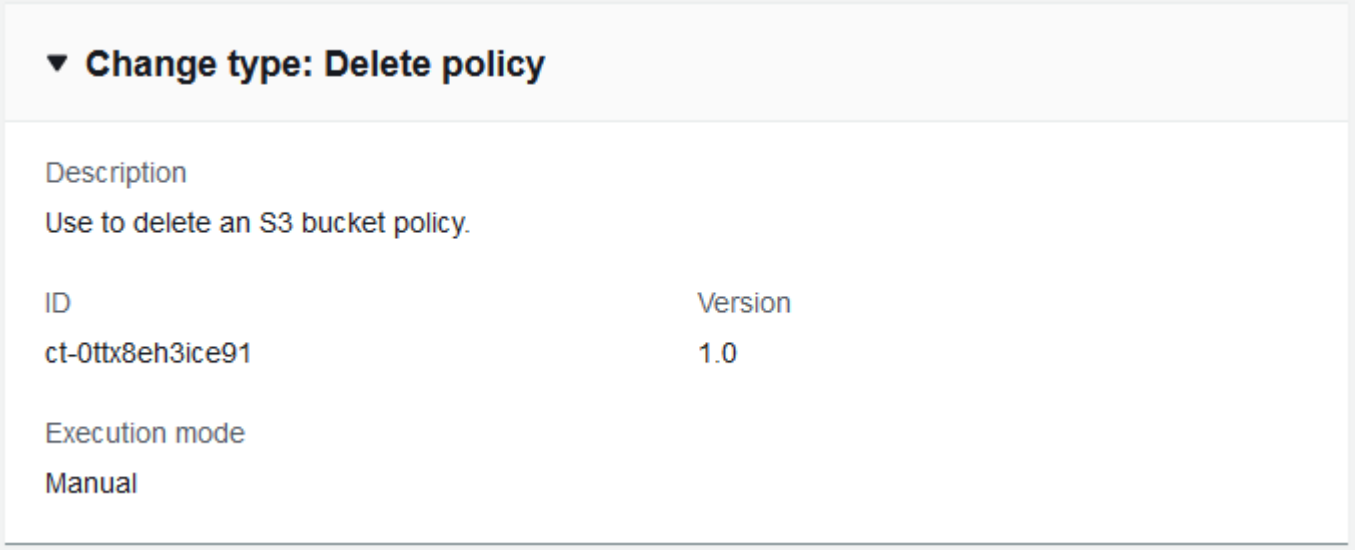
Change type ID	ct-0ttx8eh3ice91
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete S3 storage policy (review required)

Deleting an S3 Storage Policy (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an S3 Storage Policy (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0ttx8eh3ice91" --change-type-version "1.0"
 --title "TITLE" --execution-parameters "{\"BucketName\": \"example-bucket-123\",
 \"Operation\": \"Delete policy\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteS3PolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ttx8eh3ice91" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteS3PolicyParams.json
   ```

2. Modify and save the DeleteS3PolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "BucketName": "test-bucket-01",
     "Operation": "Delete policy"
   }
   ```

3. Output the RFC template JSON file to a file named DeleteS3PolicyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteS3PolicyRfc.json
   ```

4. Modify and save the DeleteS3PolicyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0ttx8eh3ice91",
   "Title":                "S3-Policy-Delete-RFC"
   ```

```
}
```

5. Create the RFC, specifying the DeleteS3PolicyRfc file and the DeleteS3PolicyParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteS3PolicyRfc.json  --execution-
parameters file://DeleteS3PolicyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ttx8eh3ice91.

Example: Required Parameters

```
{
  "BucketName": "examplebucketname",
  "Operation": "Delete policy"
}
```

Example: All Parameters

```
{
  "BucketName": "examplebucketname",
  "Operation": "Delete policy",
  "Priority": "Medium"
}
```

S3 Storage | Manage Lifecycle Configuration

Add a new lifecycle configuration, or replace an existing one for an Amazon S3 bucket.

Full classification: Management | Advanced stack components | S3 storage | Manage lifecycle configuration

Change Type Details

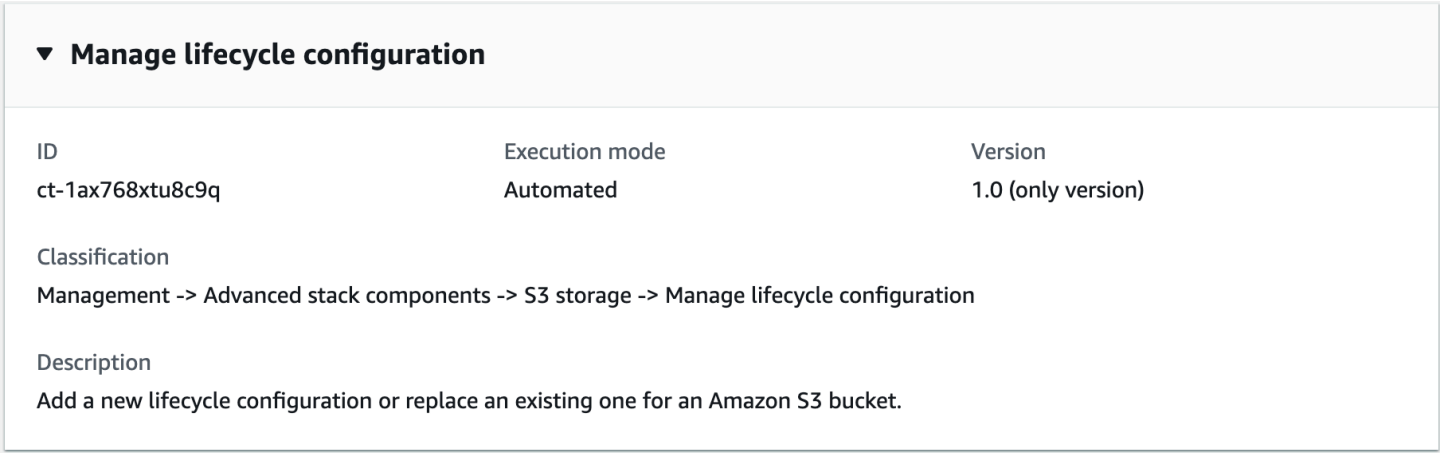
Change type ID	ct-1ax768xtu8c9q
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Manage S3 lifecycle configuration

Adding a new or replacing an existing lifecycle configuration for an S3 bucket with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

> **Note**
>
> When you add a lifecycle configuration from the console, you must provide a JSON string for the `LifecycleConfiguration` parameter similar to the following example:
> `{"Rules":[{"ID": "IDname","Filter": {"Prefix": "bucketprefix/"},"Status": "Enabled","Expiration": {"Days": 30},"NoncurrentVersionExpiration": {"NoncurrentDays": 30}}]}`

Adding a new or replacing an existing lifecycle configuration for an S3 bucket with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1ax768xtu8c9q" \
--change-type-version "1.0" --title "Manage lifecycle configuration" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
PutBucketLifecycleConfiguration\",\"Region\":\"us-east-1\",\"Parameters\":{\"BucketName
\":[\"example-bucket-123\"],\"LifecycleConfiguration\":[\"{\\\"Rules\\\":[{\\
```

```
\"Filter\\\":{\\\"Prefix\\\":\\\"documents/\\\"},\\\"Status\\\":\\\"Enabled\\\",\\
\"Transitions\\\":[{\\\"Days\\\":365,\\\"StorageClass\\\":\\\"GLACIER\\\"}],\\\"ID
\\\":\\\"ExampleRule\\\"}]}\"], \"ReplaceExisting\": [\"True\"], \"Verification\":
 [\"confirm\"], \"MinimumNumberOfDaysBeforeExpiration\": [2]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ManageS3LifecycleConfigParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-220bdb8blaixf"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ManageS3LifecycleConfigParams.json
   ```

2. Modify and save the ManageS3LifecycleConfigParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-PutBucketLifecycleConfiguration",
     "Region": "us-east-1",
     "Parameters": {
       "BucketName": ["example-bucket-123"],
       "LifecycleConfiguration": ["{\"Rules\":[{\"Filter\":{\"Prefix\":\"documents/
   \"},\"Status\":\"Enabled\",\"Transitions\":[{\"Days\":365,\"StorageClass\":
   \"GLACIER\"}],\"ID\":\"ExampleRule\"}]}"],
       "ReplaceExisting": ["True"],
       "Verification": ["confirm"],
       "MinimumNumberOfDaysBeforeExpiration": [2]
     }
   }
   ```

3. Output the RFC template JSON file to a file named ManageS3LifecycleConfigRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ManageS3LifecycleConfigRfc.json
   ```

4. Modify and save the ManageS3LifecycleConfigRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId": "ct-1ax768xtu8c9q",
   "ChangeTypeVersion": "1.0",
   "Title": "Testing - ct-1ax768xtu8c9q Manage lifecycle configuration"
   ```

```
}
```

5. Create the RFC, specifying the ManageS3LifecycleConfigRfc file and the ManageS3LifecycleConfigParams file:

```
aws amscm create-rfc --cli-input-json file://ManageS3LifecycleConfigRfc.json  --
execution-parameters file://ManageS3LifecycleConfigParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1ax768xtu8c9q.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-PutBucketLifecycleConfiguration",
  "Region" : "us-east-1",
  "Parameters" : {
    "BucketName" : [
      "test-s3-bucket"
    ],
    "LifecycleConfiguration" : [
      "{\"Rules\":[{\"Filter\":{\"Prefix\":\"documents/\"},\"Status\":\"Enabled\",
\"Transitions\":[{\"Days\":365,\"StorageClass\":\"GLACIER\"}],\"Expiration\":{\"Days
\":3650},\"ID\":\"ExampleRule\"}]}"
    ],
    "Verification" : [
      "confirm"
    ],
    "MinimumNumberOfDaysBeforeExpiration" : [
      10
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-PutBucketLifecycleConfiguration",
  "Region" : "us-east-1",
  "Parameters" : {
    "BucketName" : [
      "test-s3-bucket"
    ],
    "LifecycleConfiguration" : [
      "{\"Rules\":[{\"Filter\":{\"Prefix\":\"documents/\"},\"Status\":\"Enabled\",
\"Transitions\":[{\"Days\":365,\"StorageClass\":\"GLACIER\"}],\"Expiration\":{\"Days
\":3650},\"ID\":\"ExampleRule\"}]}"
    ],
    "ReplaceExisting" : [
      "False"
    ],
    "Verification" : [
      "confirm"
    ],
    "MinimumNumberOfDaysBeforeExpiration" : [
      10
    ]
  }
}
```

S3 Storage | Receive Replication Replica

Receive S3 object replicas in the destination bucket.

Full classification: Management | Advanced stack components | S3 storage | Receive replication replica

Change Type Details

Change type ID	ct-00zr0b0ozlcn3
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required

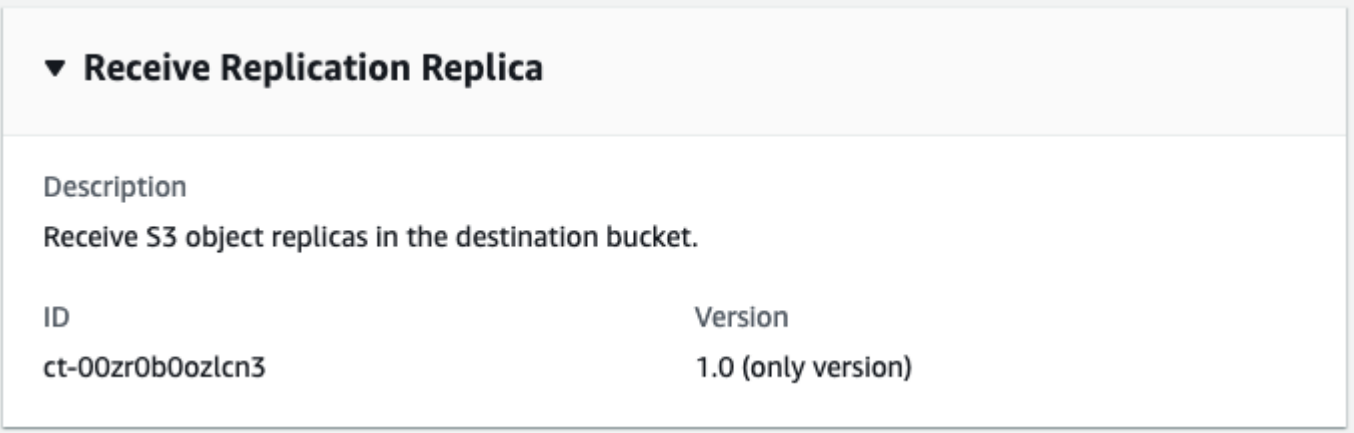
Customer approval	Not required
Execution mode	Automated

Additional Information

Receive replication replica

Receiving replication replicas in an Amazon S3 bucket using the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Receiving replication replicas in an Amazon S3 bucket using the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

With required parameters for one rule:

```
aws amscm create-rfc --change-type-id "ct-00zr0b0ozlcn3" --change-type-version
 "1.0" --title "Receive S3 object replicas in the destination bucket"--execution-
parameters"{\"DocumentName\":\"AWSManagedServices-ReceiveReplicationReplica\",
\"Region\":\"us-east-1\",\"Parameters\":{\"DestinationBucketName\":[\"destination-
bucket-name\"],\"SourceBucketName\":[\"source-bucket-name\"],\"ReplicationRole
\":[\"arn:aws:iam::123456789012:role/s3crr_role_for_test-replication\"],
\"EncryptReplicaKMSKey\":[\"arn:aws:kms:us-east-1:123456789012:key/12345678-aaaa-bbbb-
cccc-123456789012\"],\"OwnerTranslation\":[\"false\"]}}"
```

TEMPLATE CREATE:

1. Create and save the ReceiveReplicationReplicaParams.json file:

```
aws amscm get-change-type-version --change-type-id "ct-00zr0b0ozlcn3"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 ReceiveReplicationReplicaParams.json
```

2.

```
{
    "DocumentName" : "AWSManagedServices-ReceiveReplicationReplica",
    "Region": "us-east-1",
    "Parameters": {
      "DestinationBucketName" : "destination-bucket-name",
      "SourceBucketName" : "source-bucket-name",
      "ReplicationRole" : "arn:aws:iam::123456789012:role/s3crr_role_for_test-
replication",
      "EncryptReplicaKMSKey" : "arn:aws:kms:us-east-1:123456789012:key/12345678-
aaaa-bbbb-cccc-123456789012",
      "OwnerTranslation" : "false"
    }
```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it ReceiveReplicationReplicaRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ReceiveReplicationReplicaRfc.json
```

4. Modify and save the ReceiveReplicationReplicaRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeVersion": "1.0",
    "ChangeTypeId": "ct-00zr0b0ozlcn3",
    "Title": "Receive S3 object replicas in the destination bucket."
}
```

5. Create the RFC, specifying the ReceiveReplicationReplicaRfc file and the ReceiveReplicationReplicaParams file:

```
aws amscm create-rfc --cli-input-json file://ReceiveReplicationReplicaRfc.json --
execution-parameters file://ReceiveReplicationReplicaParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This is a new change type that enables you to receive replication replicas in your Amazon S3 bucket. If you want to add replication rules, use the S3 storage: Add replication replica change type.

To learn more about Replication, see Replication.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-00zr0b0ozlcn3.

Example: Required Parameters

```
{
    "DocumentName" : "AWSManagedServices-ReceiveReplicationReplica",
    "Region": "us-east-1",
    "Parameters": {
      "DestinationBucketName": ["s3-replication-destination"],
      "SourceBucketName": ["test-s3-replication"],
      "ReplicationRole": ["arn:aws:iam::555555555555:role/service-role/
s3_role_for_test-replication"]
    }
}
```

Example: All Parameters

```
{
    "DocumentName" : "AWSManagedServices-ReceiveReplicationReplica",
    "Region": "us-east-1",
    "Parameters": {
      "DestinationBucketName": ["s3-test-destination"],
      "SourceBucketName": ["s3-test-source"],
      "ReplicationRole": ["arn:aws:iam::555555555555:role/service-role/
s3_role_for_test-replication"],
      "EncryptReplicaKMSKey":["arn:aws:kms:us-
east-1:123456789012:key/12345678-5555-4375-91f5-1232d682129a"],
      "OwnerTranslation":["true"]
    }
}
```

S3 Storage | Update

Modify the properties of an S3 bucket created using change type ID ct-1a68ck03fn98r, version 4.0.

Full classification: Management | Advanced stack components | S3 storage | Update

Change Type Details

Change type ID	ct-1gi93jhvj28eg
Current version	4.0

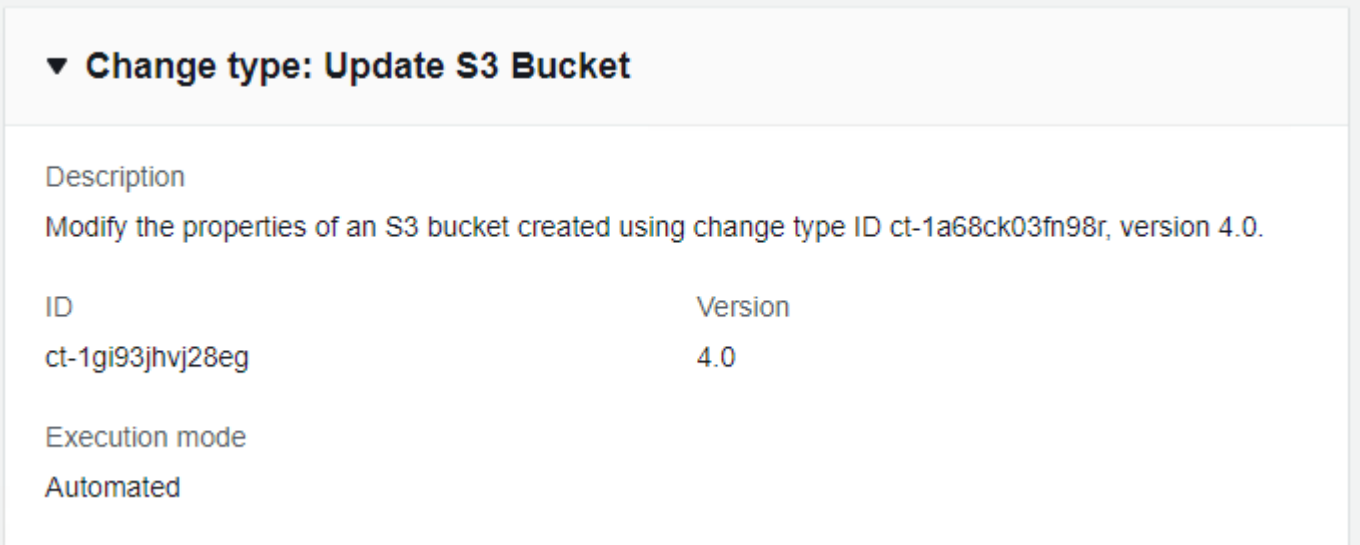
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update S3 storage

Updating an S3 with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an S3 with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Example with only required parameters:

```
aws amscm create-rfc --title s3-bucket-update --change-type-
id ct-1gi93jhvj28eg --change-type-version 4.0 --execution-
parameters '{"VpcId":"VPC_ID","StackId":"STACK_ID","Parameters":
{"IAMPrincipalsRequiringWriteObjectAccess":["arn:aws:iam::123456789012:role/roleA"]}}'
```

Example with all parameters:

```
aws amscm create-rfc --title s3-bucket-update --change-type-
id ct-1gi93jhvj28eg --change-type-version 4.0 --execution-
parameters '{"VpcId":"VPC_ID","StackId":"STACK_ID","Parameters":
{"ServerSideEncryption":"KmsManagedKeys","KMSKeyId":"arn:aws:kms:us-
east-1:123456789012:key/a5984de7-3cde-4817-
a398-92d57a8d0880","Versioning":"Enabled","IAMPrincipalsRequiringReadObjectAccess":
["arn:aws:iam::123456789012:role/roleA"],"IAMPrincipalsRequiringWriteObjectAccess":
["arn:aws:iam::123456789012:role/roleA"],"ServicesRequiringReadObjectAccess":
["logs.us-east-1.amazonaws.com"],"ServicesRequiringWriteObjectAccess":["logs.us-
east-1.amazonaws.com"],"EnforceSecureTransport":true,"AccessAllowedIpRanges":
["1.2.3.4/24"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateBucketParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-1gi93jhvj28eg" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateBucketParams.json
```

2. Modify and save the UpdateBucketParams file.

   Example with all parameters (at least one parameter must be specified):

```
{
  "VpcId": "VPC_ID",
  "StackId": "STACK_ID",
  "Parameters": {
    "ServerSideEncryption": "KmsManagedKeys",
    "KMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/a5984de7-3cde-4817-
a398-92d57a8d0880",
    "Versioning": "Enabled",
    "IAMPrincipalsRequiringReadObjectAccess": [
      "arn:aws:iam::123456789012:role/roleA"
    ],
    "IAMPrincipalsRequiringWriteObjectAccess": [
      "arn:aws:iam::123456789012:role/roleA"
    ],
    "ServicesRequiringReadObjectAccess": [
      "logs.us-east-1.amazonaws.com"
    ],
    "ServicesRequiringWriteObjectAccess": [
      "logs.us-east-1.amazonaws.com"
    ],
    "EnforceSecureTransport": true,
    "AccessAllowedIpRanges": [
      "1.2.3.4/24"
    ]
  }
}
```

   Example with required parameters (at least one parameter must be specified):

```
{
  "VpcId": "VPC_ID",
  "StackId": "STACK_ID",
  "Parameters": {
    "IAMPrincipalsRequiringWriteObjectAccess": [
      "arn:aws:iam::123456789012:role/roleA"
```

```
    ]
  }
}
```

For examples of resulting policies, see S3 Storage Bucket Create Resulting Policies.

3. Output the RFC template JSON file to a file named UpdateBucketRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateBucketRfc.json
```

4. Modify and save the UpdateBucketRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "4.0",
"ChangeTypeId":         "ct-1gi93jhvj28eg",
"Title":                "S3-Bucket-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateBucketRfc file and the UpdateBucketParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateBucketRfc.json  --execution-
parameters file://UpdateBucketParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the S3 bucket or load objects to it, look in the execution output: Use the `stack_id` to view the bucket in the Cloud Formation Console, use the **S3BucketName** to view the bucket in the S3 Console.

Tips

> **Note**
>
> This walkthrough describes, and provides example commands for, updating an AWS S3 storage bucket that was created with version 4.0 of the S3 storage Create change type (ct-1a68ck03fn98r). In that version of that change type, the **AccessControl** parameter was removed and replaced with specific parameters to allow specified services or IAM roles read or write access.

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1gi93jhvj28eg.

Example: Required Parameters

```
{
  "VpcId": "vpc-12345678901234567",
  "StackId": "stack-1234567890abcdefg",
  "Parameters": {
  }
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678",
  "StackId": "stack-1234567890abcdefg",
  "Parameters": {
    "ServerSideEncryption": "KmsManagedKeys",
    "KMSKeyId": "arn:aws:kms:ap-southeast-2:123456789012:key/9d5948f1-2082-4c07-a183-
eb829b8d81c4",
    "Versioning": "Enabled",
    "IAMPrincipalsRequiringReadObjectAccess": [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "IAMPrincipalsRequiringWriteObjectAccess": [
      "arn:aws:iam::123456789012:user/myuser",
      "arn:aws:iam::123456789012:role/myrole"
    ],
    "ServicesRequiringReadObjectAccess": [
      "rds.amazonaws.com",
      "ec2.amazonaws.com",
      "logs.ap-southeast-2.amazonaws.com"
    ],
    "ServicesRequiringWriteObjectAccess": [
      "rds.amazonaws.com",
      "ec2.amazonaws.com",
      "logs.ap-southeast-2.amazonaws.com"
```

```
    ],
    "EnforceSecureTransport": true,
    "AccessAllowedIpRanges": [
      "1.0.0.0/24",
      "2.0.0.0/24"
    ]
  }
}
```

S3 Storage | Update Encryption

Enable or update S3 bucket encryption setting through direct API calls. The S3 bucket can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-1gi93jhvj28eg instead, or ct-361tlo1k7339x if the S3 bucket was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | S3 storage | Update encryption

Change Type Details

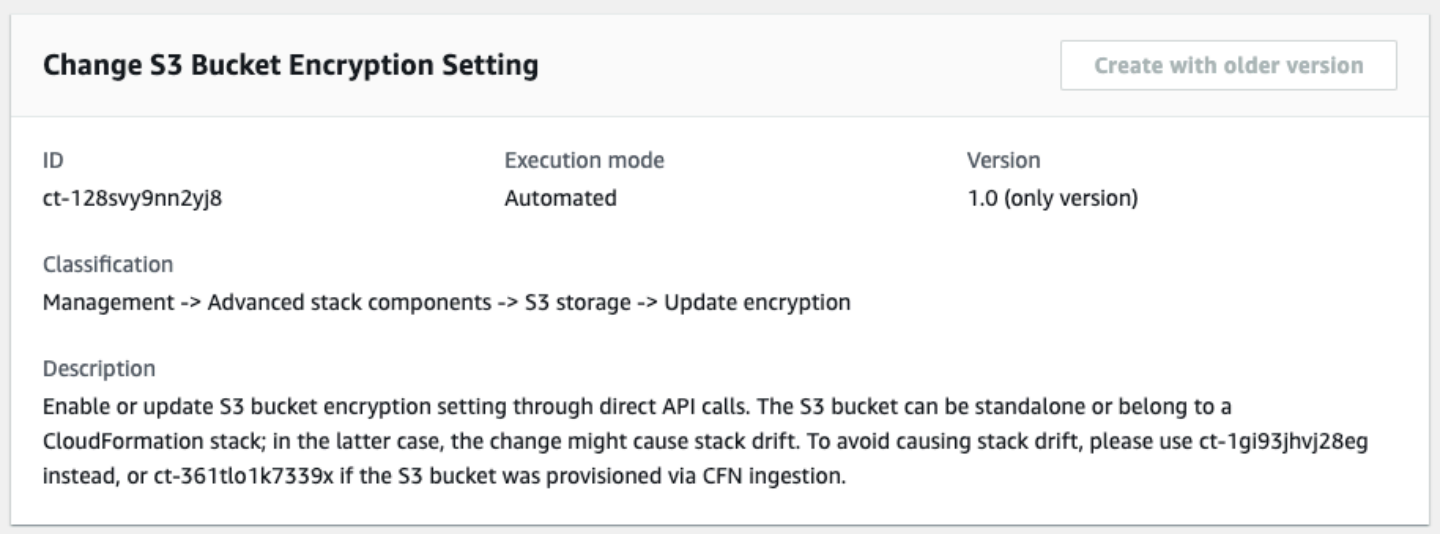
Change type ID	ct-128svy9nn2yj8
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update S3 bucket encryption

Updating S3 bucket encryption with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating S3 bucket encryption with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-128svy9nn2yj8" --change-type-version
 "1.0" --title "Update bucket encryption" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateBucketEncryption\",\"Region\":\"us-east-1\",\"Parameters
\":{\"BucketName\":[\"BucketName\"],\"ServerSideEncryption\": \"KmsManagedKeys\",
\"KMSKeyId\":[\"01234567-abcd-abcd-abcd-0123456789ab\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateBucketEncryptionParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-128svy9nn2yj8"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateBucketEncryptionParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateBucketEncryption",
     "Region": "us-east-1",
     "Parameters": {
       "BucketName": [
         "BucketName"
       ],
       "ServerSideEncryption": "KmsManagedKeys",
       "KMSKeyId": [
         "01234567-abcd-abcd-abcd-0123456789ab"
       ]
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateBucketEncryptionRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateBucketEncryptionRfc.json
   ```

4. Modify and save the UpdateBucketEncryptionRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-128svy9nn2yj8",
"Title":                "Update bucket encryption"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateBucketEncryptionRfc.json --
execution-parameters file://UpdateBucketEncryptionParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-128svy9nn2yj8.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateBucketEncryption",
  "Region": "us-east-1",
  "Parameters": {
    "BucketName": [
      "rt123456789"
    ],
    "ServerSideEncryption": "S3ManagedKeys"
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateBucketEncryption",
  "Region": "us-east-1",
  "Parameters": {
    "BucketName": [
      "rt123456789"
    ],
    "ServerSideEncryption": "KmsManagedKeys",
    "KMSKeyId": [
      "1234abcd-12ab-34cd-56ef-1234567890ab"
    ]
  }
}
```

S3 Storage | Update Policy (Review Required)

Update an S3 bucket policy.

Full classification: Management | Advanced stack components | S3 storage | Update policy (review required)

Change Type Details

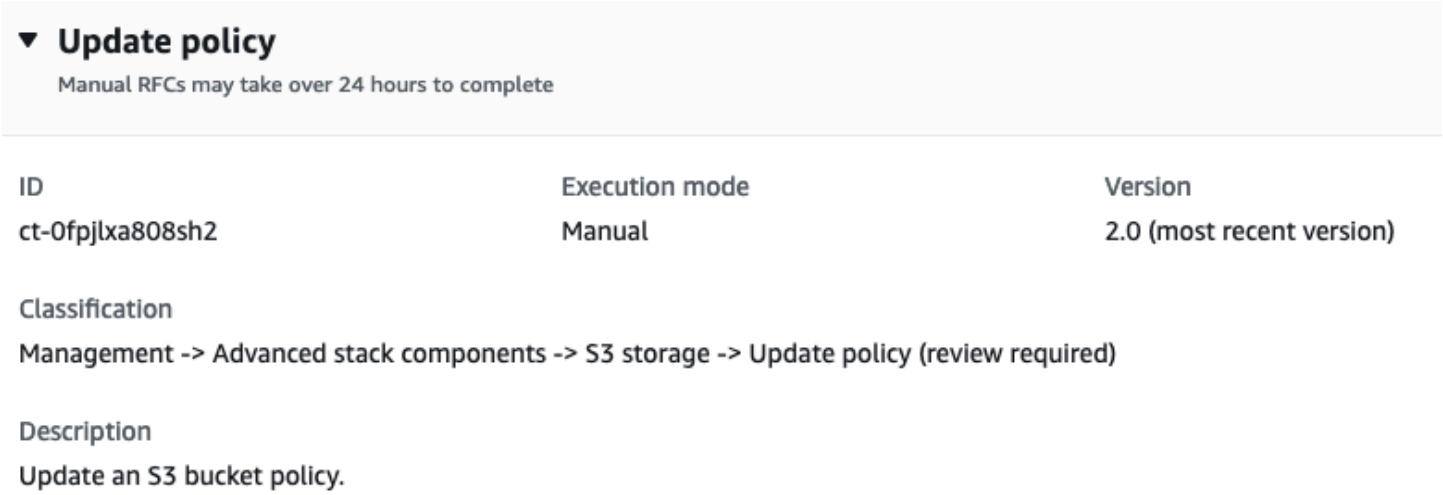
Change type ID	ct-0fpjlxa808sh2
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update S3 storage policy (review required)

Updating an S3 Storage Policy (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an S3 Storage Policy (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0fpjlxa808sh2" --change-type-version "2.0"
 --title "TITLE" --execution-parameters "{\"BucketName\": \"example-bucket-123\",
 \"BucketPolicy\": \"Example bucket policy\", \"Operation\": \"Update policy\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateS3PolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0fpjlxa808sh2" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateS3PolicyParams.json
   ```

2. Modify and save the UpdateS3PolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "BucketName": "examplebucketname",
     "BucketPolicy": "Example bucket permissions",
     "Operation": "Update policy",
     "PolicyAction": "Append"
   }
   ```

3. Output the RFC template JSON file to a file named UpdateS3PolicyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateS3PolicyRfc.json
   ```

4. Modify and save the UpdateS3PolicyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "2.0",
   "ChangeTypeId":         "ct-0fpjlxa808sh2",
   "Title":                "S3-Policy-Update-RFC"
   }
   ```

5. Create the RFC, specifying the UpdateS3PolicyRfc file and the UpdateS3PolicyParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateS3PolicyRfc.json  --execution-
parameters file://UpdateS3PolicyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0fpjlxa808sh2.

Example: Required Parameters

```
{
  "BucketName": "examplebucketname",
  "BucketPolicy": "Example bucket permissions",
  "PolicyAction": "Append",
  "Operation": "Update policy"
}
```

Example: All Parameters

```
{
  "BucketName": "examplebucketname",
  "BucketPolicy": "Example bucket permissions",
  "PolicyAction": "Append",
  "Operation": "Update policy",
  "Priority": "Medium"
}
```

S3 Storage | Update Versioning

Change S3 bucket versioning setting through direct API calls. The S3 bucket can be standalone or belong to a CloudFormation stack; in the latter case, the change might cause stack drift. To avoid causing stack drift, please use ct-1gi93jhvj28eg instead, or ct-361tlo1k7339x if the S3 bucket was provisioned via CFN ingestion.

Full classification: Management | Advanced stack components | S3 storage | Update versioning

Change Type Details

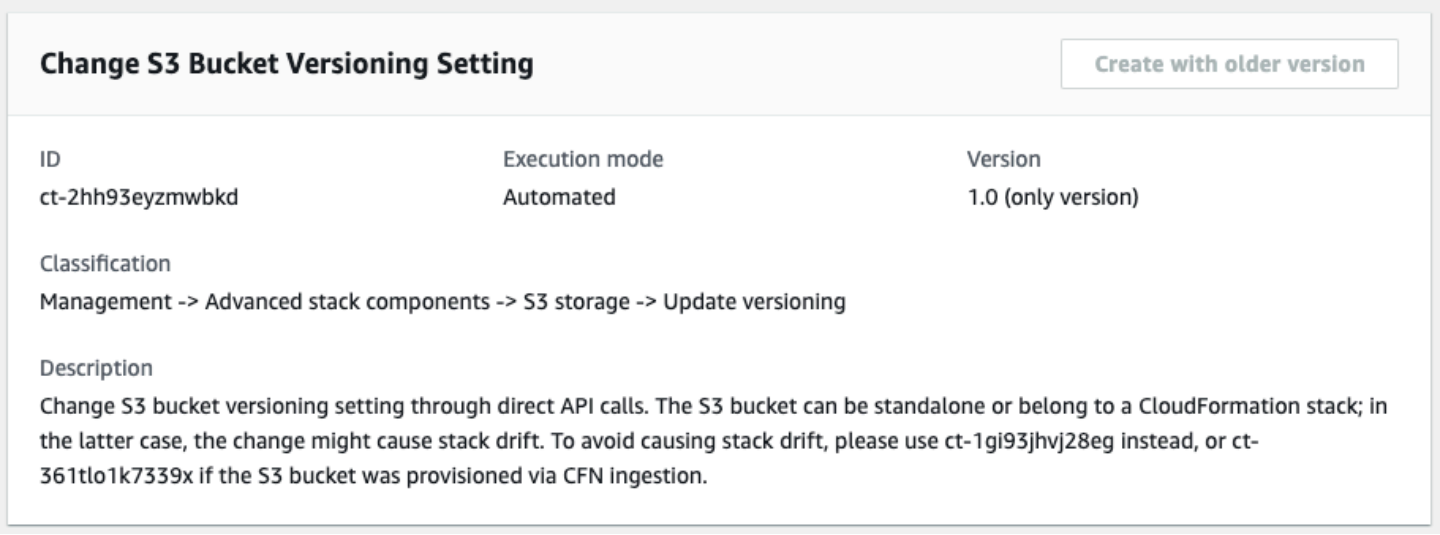
Change type ID	ct-2hh93eyzmwbkd
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update S3 bucket versioning

Updating S3 bucket versioning with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating S3 bucket versioning with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2hh93eyzmwbkd" --change-type-version
 "1.0" --title "Update bucket versioning" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateBucketVersioning\",\"Region\":\"us-east-1\",\"Parameters\":
{\"BucketName\":[\"BucketName\"],\"Versioning\": \"Enabled\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named UpdateBucketVersioningParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2hh93eyzmwbkd"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateBucketVersioningParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateBucketVersioning",
     "Region": "us-east-1",
     "Parameters": {
       "BucketName": [
         "BucketName"
       ],
       "Versioning": "Enabled"
     }
   }
   ```

3. Output the JSON template to a file in your current folder; this example names it UpdateBucketVersioningRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateBucketVersioningRfc.json
   ```

4. Modify and save the UpdateBucketVersioningRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2hh93eyzmwbkd",
"Title":                "Update bucket versioning"
}
```

5. Create the RFC, specifying the execution parameters file and the UpdateRdsRfc file:

```
aws amscm create-rfc --cli-input-json file://UpdateBucketVersioningRfc.json --
execution-parameters file://UpdateBucketVersioningParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon S3, see Amazon Simple Storage Service Documentation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2hh93eyzmwbkd.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateBucketVersioning",
  "Region": "us-east-1",
  "Parameters": {
    "BucketName": [
      "rt123456789"
    ],
    "Versioning": "Enabled"
  }
}
```

Security Group | Associate

Associate security groups with an AWS resource.

Full classification: Management | Advanced stack components | Security group | Associate

Change Type Details

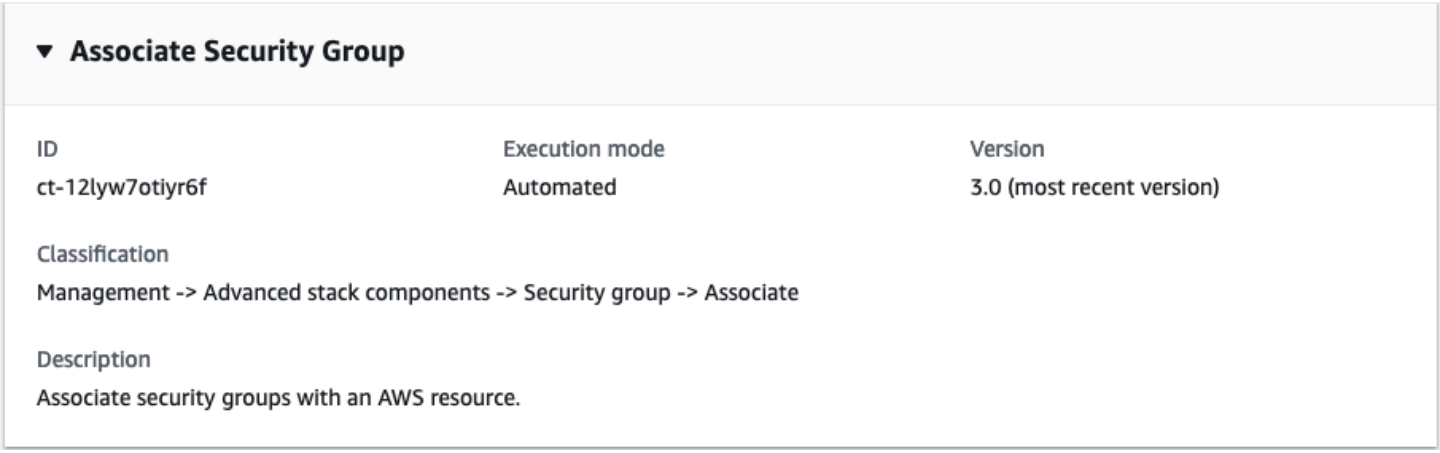
Change type ID	ct-12lyw7otiyr6f
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Associate security group to resource

Associating a Security Group to Resources with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Associating a Security Group to Resources with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-12lyw7otiyr6f" --change-type-version "3.0"
 --title "Associate Security Groups" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-AttachSecurityGroupsV2\", \"Region\": \"us-east-1\", \"Parameters
\": {\"ResourceType\": \"EC2Instance\", \"ResourceId\": \"i-xxxxxxxxxxxxxxxxx\",
 \"SecurityGroupIds\": [\"sg-xxxxxxxxxxxxxxxxx\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AssociateSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-12lyw7otiyr6f" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AssociateSGParams.json
   ```

2. Modify and save the AssociateSGParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-AttachSecurityGroupsV2",
       "Region": "us-east-1",
   ```

```
    "Parameters": {
        "ResourceType": [
            "EC2Instance"
        ],
        "ResourceId": [
            "i-xxxxxxxxxxxxxxxxx"
        ],
        "SecurityGroupIds": [
            "sg-xxxxxxxxxxxxxxxxx"
        ]
    }
}
```

3. Output the RFC template JSON file to a file named AssociateSGRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AssociateSGRfc.json
```

4. Modify and save the AssociateSGRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "3.0",
"ChangeTypeId":         "ct-12lyw7otiyr6f",
"Title":                "SG-Associate-RFC"
}
```

5. Create the RFC, specifying the AssociateSG Rfc file and the AssociateSGParams file:

```
aws amscm create-rfc --cli-input-json file://AssociateSGRfc.json  --execution-
parameters file://AssociateSGParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 3.0. The schema is changed to use an SSM document and new resource types are now supported.

Important

For AutoScalingGroupCurrentInstancesOnly, security groups are only attached to individual instances currently part of the ASG. LaunchTemplate or LaunchConfiguration are not updated. Bes sure to update LaunchTemplate / LaunchConfiguration before updating security groups to AutoScalingGroup Instances.

Important

If true, any access allowed by existing security groups is removed and only the new security groups are in effect.

To learn more about associating security groups to resources, see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-12lyw7otiyr6f.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-AttachSecurityGroupsV2",
    "Region": "us-east-1",
    "Parameters": {
      "ResourceType": "RDSDBInstance",
      "ResourceId": "MyDBInstance",
      "SecurityGroupIds": ["sg-1234556eaba0a4799"],
      "OverwriteSecurityGroups": "true"
    }
}
```

Security Group | Authorize Egress Rule

Authorize the egress rule for the specified security group (SG). You must specify the configurations of the egress rule that you are authorizing. Note that this adds an egress rule to the specified SG but does not modify any existing egress rules.

Full classification: Management | Advanced stack components | Security group | Authorize egress rule

Change Type Details

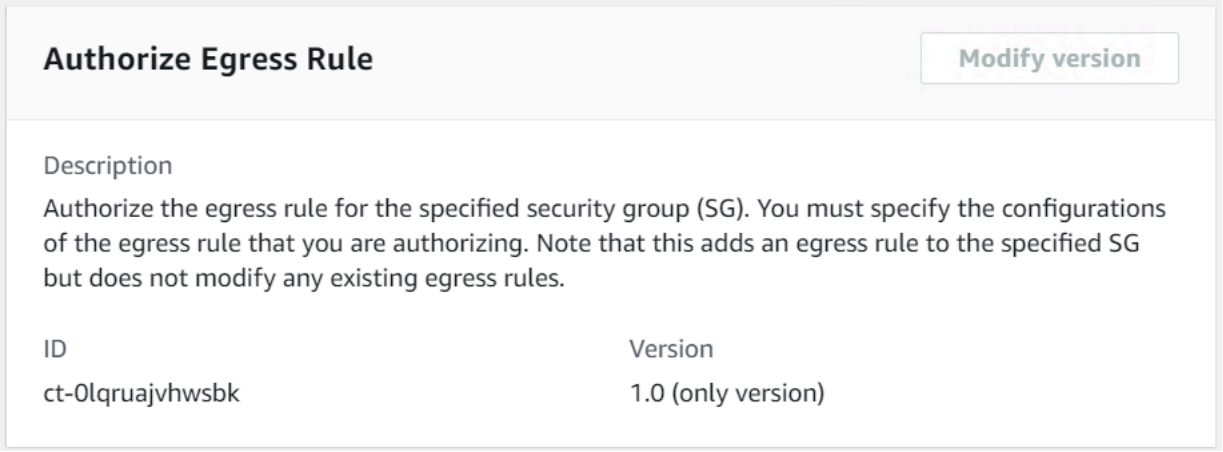
Change type ID	ct-0lqruajvhwsbk
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Authorize security group egress rule

Authorizing a security group egress rule with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Authorizing a security group egress rule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0lqruajvhwsbk" --change-type-version
 "1.0" --title "Authorize security group egress rule" --execution-parameters
 '{"DocumentName":"AWSManagedServices-AuthorizeSecurityGroupEgressRule","Region":"us-
east-1","Parameters":{"SecurityGroupId":["SG_ID"],"IpProtocol":["tcp"],"FromPort":
[80],"ToPort":[80],"Destination":["10.0.0.1/24"],"Description":["HTTP Port for
 10.0.0.1/24"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AuthSGEgressParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0lqruajvhwsbk" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AuthSGEgressParams.json
   ```

2. Modify and save the AuthSGEgressParams file. For example, you can replace the contents with something like this:

```
{
"DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupEgressRule",
"Region" : "us-east-1",
"Parameters" : {
"SecurityGroupId" : ["SG_ID"],
"IpProtocol" : ["tcp"],
"FromPort" : [80],
"ToPort" : [80],
"Destination" : ["10.0.0.1/24"]
"Description" : ["HTTP Port for 10.0.0.1/24"]
}
}
```

3. Output the RFC template JSON file to a file named AuthSGEgressRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AuthSGEgressRfc.json
```

4. Modify and save the AuthSGEgressRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-0lqruajvhwsbk",
  "ChangeTypeVersion": "1.0",
  "Title": "Authorize security group egress rule"
}
```

5. Create the RFC, specifying the AuthSGEgressRfc file and the AuthSGEgressParams file:

```
aws amscm create-rfc --cli-input-json file://AuthSGEgressRfc.json  --execution-
parameters file://AuthSGEgressParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> There are two ways to authorize a new egress rule, one, Security Group: Update change type (ct-3memthlcmvc1b), has ExecutionMode=Manual and provides a lot of latitude for custom rules; however, being manual, it takes longer to execute as AMS Operations must review it for safety, and possibly require communications. The other egress rule authorization way, Security Group: Authorize Egress Rule change type (ct-3j2zstluz6dxq), has ExecutionMode=Automated and provides options for creating standard TCP/UDP or ICMP egress rules. It is more limited in scope; however, being automated, it executes more quickly.
> This walkthrough is for the Security Group: Authorize Egress Rule change type.

To learn more about AWS security groups and security group rules, see Security Group Rules Reference; this page can help you determine the rules you want and, importantly, how to name your security group so choosing it when creating other resources is intuitive. Also see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

Once the security group is created, use Associate security group to resource to associate the security group with your AMS resources. In order to delete a security group, it must have associated resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0lqruajvhwsbk.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupEgressRule",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
```

```
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Destination" : [
      "10.0.0.1/32"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupEgressRule",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Destination" : [
      "10.0.0.1/32"
    ],
    "Description" : [
      "New rule"
    ]
  }
}
```

Security Group | Authorize Ingress Rule

Authorize the ingress rule for the specified security group (SG). You must specify the configurations of the ingress rule that you are authorizing. Note that this adds an ingress rule to the specified SG but does not modify any existing ingress rules.

Full classification: Management | Advanced stack components | Security group | Authorize ingress rule

Change Type Details

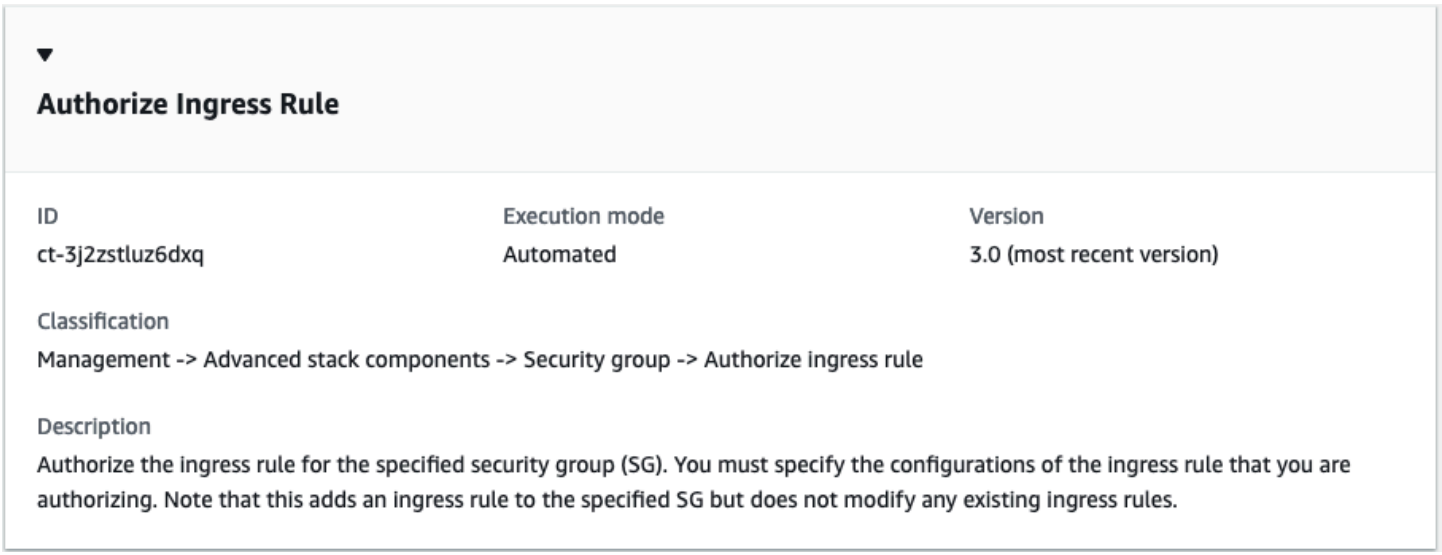
Change type ID	ct-3j2zstluz6dxq
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Authorize security group ingress rule

Authorizing a security group ingress rule with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Authorizing a security group ingress rule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3j2zstluz6dxq" --change-
type-version "3.0" --title "Authorize security group ingress rule"
 --execution-parameters '{"DocumentName":"AWSManagedServices-
AuthorizeSecurityGroupIngressRuleV3","Region":"us-east-1","Parameters":
{"SecurityGroupId":["SG_ID"],"IpProtocol":["tcp"],"FromPort":[80],"ToPort":
[80],"Source":["10.0.0.1/24"],"Description":["HTTP Port for 10.0.0.1/24"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AuthSGIngressParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3j2zstluz6dxq" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AuthSGIngressParams.json
   ```

2. Modify and save the AuthSGIngressParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupIngressRuleV3",
     "Region" : "us-east-1",
     "Parameters" : {
       "SecurityGroupId" : [
         "SG_ID"
       ],
       "IpProtocol" : [
         "tcp"
       ],
       "FromPort" : [
         80
       ],
       "ToPort" : [
         80
       ],
       "Source" : [
         "10.0.0.1/24"
       ],
       "Description" : [
   ```

```
      "HTTP Port for 10.0.0.1/24"
    ]
  }
}
```

3. Output the RFC template JSON file to a file named AuthSGIngressRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AuthSGIngressRfc.json
```

4. Modify and save the AuthSGIngressRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-3j2zstluz6dxq",
  "ChangeTypeVersion": "3.0",
  "Title": "Authorize security group ingress rule"
}
```

5. Create the RFC, specifying the AuthSGIngressRfc file and the AuthSGIngressParams file:

```
aws amscm create-rfc --cli-input-json file://AuthSGIngressRfc.json  --execution-
parameters file://AuthSGIngressParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Note

This change type is now at version 2.0. The two separate, optional source parameters, **CidrIp** and **SourceSecurityGroupId**, have been combined into one required parameter, **Source**, with two options. This was done to ensure that a source was provided; without a source, the RFC would fail.

There are two ways to authorize a new ingress rule: use Security Group: Update change type (ct-3memthlcmvc1b), which is a manual change type (and thus takes longer to implement because AMS Operations must review it for safety, and may require communications); alternatively, use this change type (ct-3j2zstluz6dxq), which is automated

(and thus implemented more quickly) and provides options for deleting standard TCP/UDP or ICMP ingress rules.

To learn more about AWS security groups and security group rules, see Security Group Rules Reference; this page can help you determine the rules you want and, importantly, how to name your security group so choosing it when creating other resources is intuitive. Also see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

Once the security group is created, use Associate security group to resource to associate the security group with your AMS resources. In order to delete a security group, it must have associated resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3j2zstluz6dxq.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupIngressRuleV3",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Source" : [
      "10.0.0.1/32"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AuthorizeSecurityGroupIngressRuleV3",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Source" : [
      "10.0.0.0"
    ],
    "Description" : [
      "New rule"
    ]
  }
}
```

Security Group | Delete

Delete up to 20 security groups. Note: Only security groups with no dependencies are deleted and security groups with dependencies are not deleted. This change type does not require a review and can be used instead of the manual, review required, change type (ct-3cp96z7r065e4).

Full classification: Management | Advanced stack components | Security group | Delete

Change Type Details

Change type ID	ct-18r16ldqil6w9
Current version	1.0
Expected execution duration	60 minutes

AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete security group

Deleting an AMS Security Group with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AMS Security Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) , and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-18r16ldqil6w9" --change-type-version
 "1.0" --title "Delete security group" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-DeleteSecurityGroups\", \"Region\": \"us-east-1\", \"Parameters
\": {\"SecurityGroupIds\": [\"sg-xxxxxxxxxxxxxxxxx\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-18r16ldqil6w9" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteSGParams.json
   ```

2. Modify and save the DeleteSGParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-DeleteSecurityGroups",
       "Region": "us-east-1",
       "Parameters": {
           "SecurityGroupIds": [
               "sg-xxxxxxxxxxxxxxxxx"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file named DeleteSGRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteSGRfc.json
   ```

4. Modify and save the DeleteSGRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-18r16ldqil6w9",
   "Title":                "SG-Delete-RFC"
   ```

```
}
```

5. Create the RFC, specifying the DeleteSG Rfc file and the eleteSGParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteSGRfc.json  --execution-
parameters file://DeleteSGParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> You must first separate the security group from any resources associated with it or the RFC fails. Only security groups with no dependencies are deleted and security groups with dependencies are not deleted.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-18r16ldqil6w9.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-DeleteSecurityGroups",
    "Region": "us-east-1",
    "Parameters": {
      "SecurityGroupIds": ["sg-1234556eaba0a4799"],
      "ForceDelete": ["true"]
    }
}
```

Security Group | Delete (Review Required)

Disassociate a security group from the specified AWS resources and optionally delete the security group.

Full classification: Management | Advanced stack components | Security group | Delete (review required)

Change Type Details

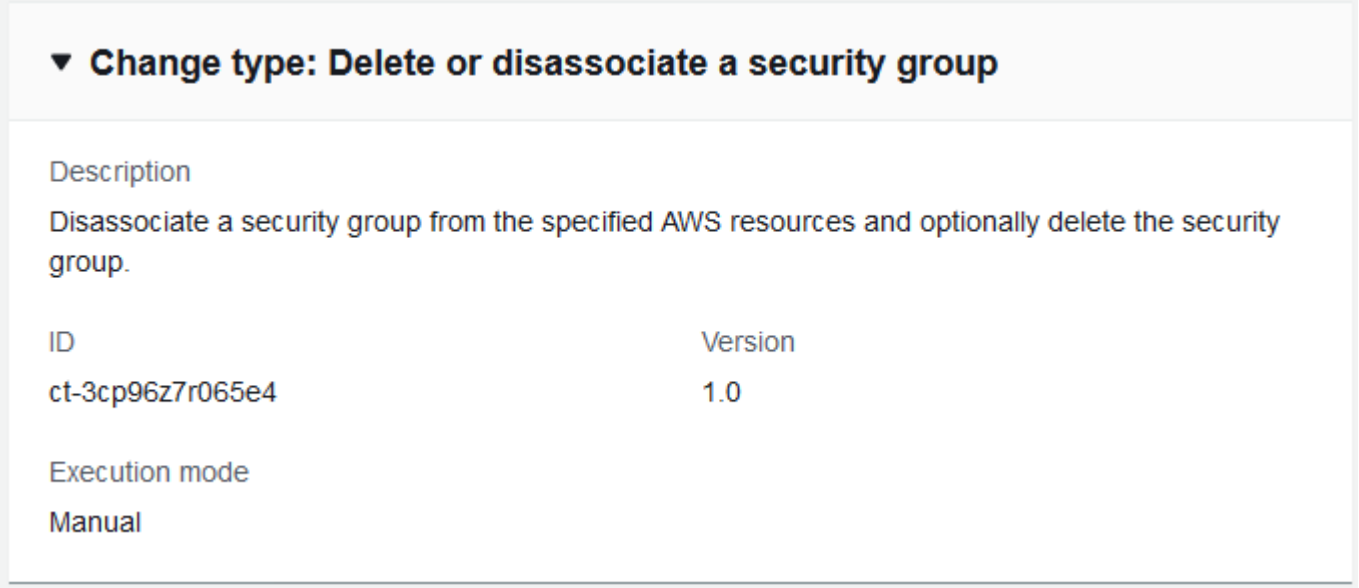
Change type ID	ct-3cp96z7r065e4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete security group (review required)

Deleting an AMS Security Group with the Console (review required)

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AMS Security Group with the CLI (review required)

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

- Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline) , and then submit the returned RFC ID. For example, you can replace the contents with something like this:

  To remove associated resources, you can issue a command similar to this that uses the Delete Security Group CT (note that `SecurityGroupID` and `DeleteSecurityGroup` are required parameters):

  ```
  aws --profile saml amscm create-rfc --change-type-id "ct-3cp96z7r065e4"
   --change-type-version "1.0" --title "Remove-SG-Resources" --execution-
  parameters "{\"SecurityGroupId\":\"SG_ID\", \"DeleteSecurityGroup\":\false,
   \"DisassociatedResources\":\"IDS_OF_RESOURCES\"}"
  ```

  (Optional) To delete the security group, you can issue a command similar to this that uses the Delete Security Group CT (note that `SecurityGroupID` and `DeleteSecurityGroup` are required parameters):

  ```
  aws --profile saml amscm create-rfc --change-type-id "ct-3cp96z7r065e4" --change-
  type-version "1.0" --title "Remove-SG" --execution-parameters "{\"SecurityGroupId
  \":\"SG_ID\", \"DeleteSecurityGroup\":\true, \"DisassociatedResources\":
  \"IDS_OF_RESOURCES\"}"
  ```

  A security group cannot be deleted until all of the associated resources have been removed; use the `DisassociatedResources` parameter in the Delete Security group CT to disassociate all of the associated resources. If all resources have been disassociated, use this `\"DisassociatedResources\":\"[]\"`.

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3cp96z7r065e4" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteSGParams.json
   ```

2. Modify and save the DeleteSGParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "SecurityGroupId": "sg-1234abcd",
  "DisassociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-567890abcdefg1234"
  ],
  "DeleteSecurityGroup": true
}
```

3. Output the RFC template JSON file to a file named DeleteSGRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteSGRfc.json
```

4. Modify and save the DeleteSGRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3cp96z7r065e4",
"Title":                "SG-Delete-RFC"
}
```

5. Create the RFC, specifying the DeleteSG Rfc file and the eleteSGParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteSGRfc.json  --execution-
parameters file://DeleteSGParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. (Optional) To add inbound or outbound rules, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b"
 --change-type-version "1.0" --title "Add-SG-Rules" --execution-parameters
 "{\"SecurityGroupId\":\"SG_ID\", \"AddInboundRules\":{\"Protocol\":\"TCP\",
 \"PortRange\":\"49152-65535\, \"Source\":\"203.0.113.5/32\"}, \"AddOutboundRules
\":{\"Protocol\":\"TCP\", \"PortRange\":\"49152-65535\, \"Destination\":
\"203.0.113.5/32\"}}"
```

7. (Optional) To remove inbound or outbound rules, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b" --
change-type-version "1.0" --title "Remove-SG-Rules" --execution-parameters
 "{\"SecurityGroupId\":\"SG_ID\", \"Name\":\"MA-Test-SG-QC\", \"RemoveInboundRules
\":{\"Protocol\":\"TCP\", \"PortRange\":\"49152-65535\, \"Source\":
\"203.0.113.5/32\"}, \"RemoveOutboundRules\":{\"Protocol\":\"TCP\", \"PortRange\":
\"49152-65535\, \"Destination\":\"203.0.113.5/32\"}}"
```

8. (Optional) To add associated resources, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b" --
change-type-version "1.0" --title "Add-SG-Resources" --execution-parameters
 "{\"SecurityGroupId\":\"SG_ID\", \"AssociatedResources\":\"IDS_OF_RESOURCES\"}"
```

9. (Optional) To remove associated resources, you can issue a command similar to this that uses the Delete Security Group CT (note that `SecurityGroupID` and `DeleteSecurityGroup` are required parameters):

```
aws --profile saml amscm create-rfc --change-type-id "ct-3cp96z7r065e4"
 --change-type-version "1.0" --title "Remove-SG-Resources" --execution-
parameters "{\"SecurityGroupId\":\"SG_ID\", \"DeleteSecurityGroup\":\false,
 \"DisassociatedResources\":\"IDS_OF_RESOURCES\"}"
```

Tips

> **Note**
>
> There is an automated change type for deleting a security group, Deployment | Advanced stack components | Security group | Delete (no review required) (ct-18r16ldqil6w9) that may execute more quickly than this change type. For details, see Delete security group.

> **Note**
>
> You must first separate the security group from any resources associated with it or the RFC fails.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3cp96z7r065e4.

Example: Required Parameters

```
{
  "SecurityGroupId": "sg-1234abcd",
  "DisassociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-34abcd12",
    "i-4abcd123",
    "i-abcd1234",
    "i-1234567890abcdefg",
    "i-234567890abcdefg1",
    "i-34567890abcdefg12",
    "i-4567890abcdefg123",
    "i-567890abcdefg1234"
  ],
  "DeleteSecurityGroup": false,
  "Priority": "Medium"
}
```

Example: All Parameters

```
{
  "SecurityGroupId": "sg-1234abcd",
  "DisassociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-34abcd12",
    "i-4abcd123",
    "i-abcd1234",
    "i-1234567890abcdefg",
```

```
    "i-234567890abcdefg1",
    "i-34567890abcdefg12",
    "i-4567890abcdefg123",
    "i-567890abcdefg1234"
  ],
  "DeleteSecurityGroup": true,
  "Priority": "Medium"
}
```

Security Group | Disassociate

Disassociate a security group from up to 50 AWS resources and optionally delete the security group. This change type does not require a review and can be used instead of the manual, review required, change type (ct-3cp96z7r065e4).

Full classification: Management | Advanced stack components | Security group | Disassociate

Change Type Details

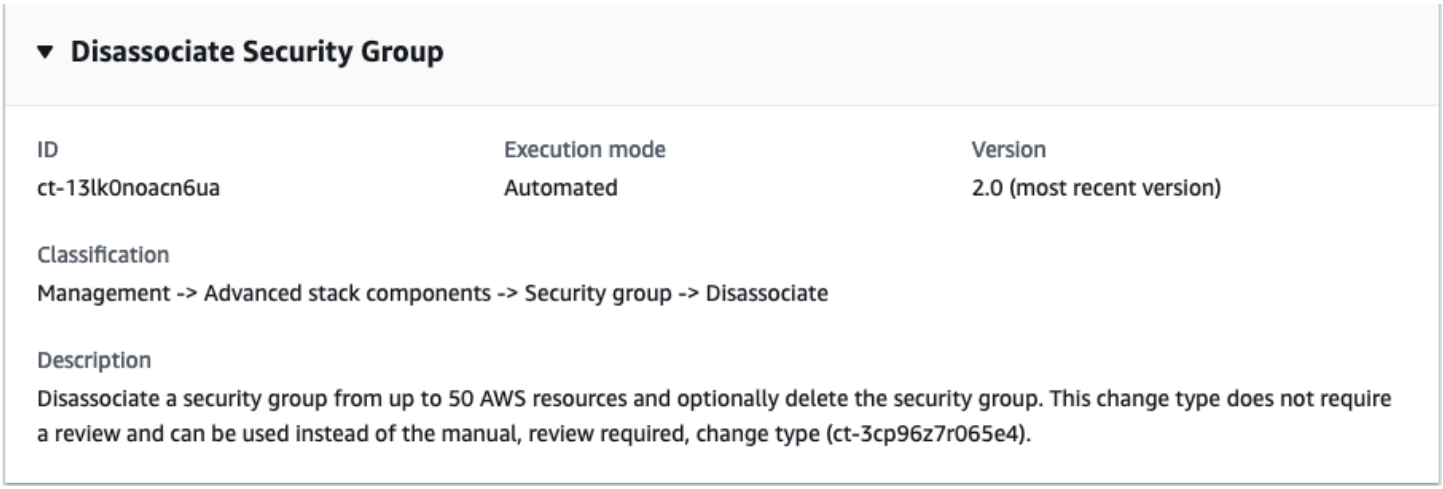
Change type ID	ct-13lk0noacn6ua
Current version	2.0
Expected execution duration	120 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Disassociate security group to resource

Disassociating a Security Group to Resources with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Disassociating a Security Group to Resources with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-13lk0noacn6ua" --change-type-version "2.0"
 --title "Disassociate security group" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-DisassociateSecurityGroupV2\", \"Region\": \"us-east-1\",
 \"Parameters\": {\"SecurityGroupId\": \"sg-xxxxxxxxxxxxxxxxx\", \"EC2InstanceIds\":
 [\"i-xxxxxxxxxxxxxxxxx\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DisassociateSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-13lk0noacn6ua" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DisassociateSGParams.json
   ```

2. Modify and save the DisassociateSGParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-DisassociateSecurityGroupV2",
       "Region": "us-east-1",
       "Parameters": {
           "SecurityGroupId": [
               "sg-xxxxxxxxxxxxxxxxx"
           ],
           "EC2InstanceIds": [
               "i-xxxxxxxxxxxxxxxxx"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file named DisassociateSGRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DisassociateSGRfc.json
   ```

4. Modify and save the DisassociateSGRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "2.0",
     "ChangeTypeId": "ct-13lk0noacn6ua",
     "Title": "Disassociate security group"
   ```

```
}
```

5. Create the RFC, specifying the DisassociateSG Rfc file and the DisassociateSGParams file:

```
aws amscm create-rfc --cli-input-json file://DisassociateSGRfc.json  --execution-
parameters file://DisassociateSGParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For help deleting a security group from a VPC, see Why can't I delete a security group for my Amazon VPC?.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-13lk0noacn6ua.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-DisassociateSecurityGroupV2",
    "Region": "us-east-1",
    "Parameters": {
      "SecurityGroupId": "sg-1234556eaba0a4799",
      "EC2InstanceIds": ["i-1234567890abababa"],
      "ElasticNetworkInterfaceIds": ["eni-1234567890abababa"],
      "AutoScalingGroupNames": ["myautoscalinggroup"],
      "ElasticLoadBalancerNames": ["myloadbalancer"],
      "ApplicationLoadBalancerNames": ["myloadbalancer"],
      "RDSDBInstanceIdentifiers": ["mydbinstance"],
      "RDSDBClusterIdentifiers": ["mydbcluster"],
      "ElasticacheClusterIdentifiers": ["mycachecluster"],
      "RedshiftClusterIdentifiers": ["myredshiftcluster"],
```

```
      "ElasticFileSystemIds": ["myfilesystem"]
    }
}
```

Security Group | Revoke Egress Rule

Revoke the egress rule for the specified security group (SG). You must specify the configurations of the egress rule that you are revoking. Note that, once revoked, the egress rule is permanently deleted.

Full classification: Management | Advanced stack components | Security group | Revoke egress rule

Change Type Details

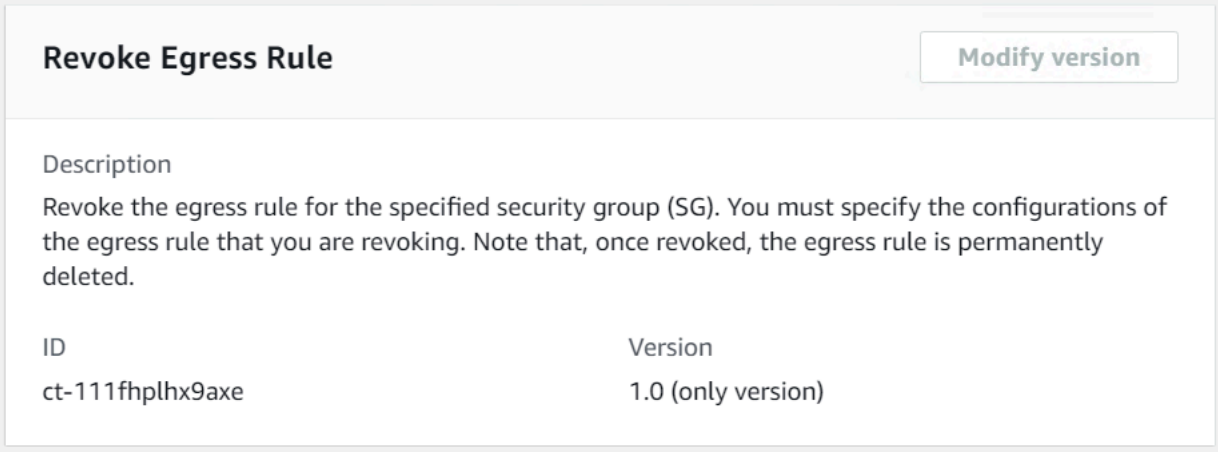
Change type ID	ct-111fhplhx9axe
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Revoke security group egress rule

Revoking a Security Group Egress Rule with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Revoking a Security Group Egress Rule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-111fhplhx9axe" --change-type-version
 "1.0" --title "Revoke security group egress rule" --execution-parameters
 '{"DocumentName":"AWSManagedServices-RevokeSecurityGroupEgressRule","Region":"us-
east-1","Parameters":{"SecurityGroupId":["SG_ID"],"IpProtocol":["tcp"],"FromPort":
[80],"ToPort":[80],"Destination":["10.0.0.1/24"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RevokeSGEgressParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-111fhplhx9axe" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RevokeSGEgressParams.json
   ```

2. Modify and save the RevokeSGEgressParams file. For example, you can replace the contents with something like this:

   ```
   {
   "DocumentName" : "AWSManagedServices-RevokeSecurityGroupEgressRule",
   "Region" : "us-east-1",
   "Parameters" : {
   "SecurityGroupId" : ["SG_ID"],
   "IpProtocol" : ["tcp"],
   "FromPort" : [80],
   "ToPort" : [80],
   "Destination" : ["10.0.0.1/24"]
   }
   }
   ```

3. Output the RFC template JSON file to a file named RevokeSGEgressRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RevokeSGEgressRfc.json
   ```

4. Modify and save the RevokeSGEgressRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeId": "ct-111fhplhx9axe",
     "ChangeTypeVersion": "1.0",
     "Title": "Revokeorize security group egress rule"
   }
   ```

5. Create the RFC, specifying the RevokeSGEgressRfc file and the RevokeSGEgressParams file:

```
aws amscm create-rfc --cli-input-json file://RevokeSGEgressRfc.json  --execution-
parameters file://RevokeSGEgressParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> There are two ways to revoke an egress rule, one, Security Group: Update change type (ct-3memthlcmvc1b), has ExecutionMode=Manual; being manual, it takes longer to execute as AMS Operations must review it for safety, and possibly require communications. The other egress rule revoke way, Security Group: Revoke Egress Rule change type (ct-1vjbacfr4ufdv), has ExecutionMode=Automated and provides options for deleting standard TCP/UDP or ICMP egress rules. It is more limited in scope; however, being automated, it executes more quickly.
> This walkthrough is for the Security Group: Revoke Egress Rule change type.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-111fhplhx9axe.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-RevokeSecurityGroupEgressRule",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
```

```
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Destination" : [
      "10.0.0.1/32"
    ]
  }
}
```

Security Group | Revoke Ingress Rule

Revoke the ingress rule for the specified security group (SG). You must specify the configurations of the ingress rule that you are revoking. Note that, once revoked, the ingress rule is permanently deleted.

Full classification: Management | Advanced stack components | Security group | Revoke ingress rule

Change Type Details

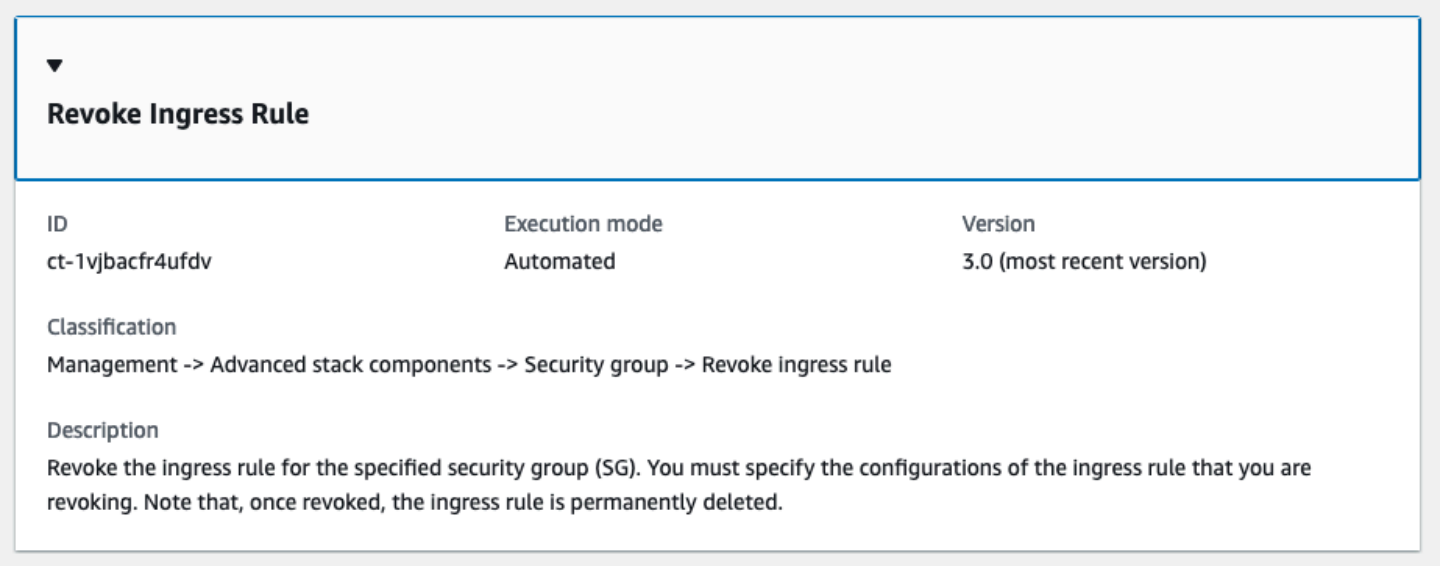
Change type ID	ct-1vjbacfr4ufdv
Current version	3.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Revoke security group ingress rule

Revoking a Security Group Ingress Rule with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Revoking a Security Group Ingress Rule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1vjbacfr4ufdv" --change-type-version
 "3.0" --title "Revoke security group ingress rule" --execution-parameters
 '{"DocumentName":"AWSManagedServices-RevokeSecurityGroupIngressRuleV3","Region":"us-
east-1","Parameters":{"SecurityGroupId":["SG_ID"],"IpProtocol":["tcp"],"FromPort":
[80],"ToPort":[80],"Source":["10.0.0.1/24"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RevokeSGIngressParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vjbacfr4ufdv"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RevokeSGIngressParams.json
   ```

2. Modify and save the RevokeSGIngressParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-RevokeSecurityGroupIngressRuleV3",
     "Region" : "us-east-1",
     "Parameters" : {
       "SecurityGroupId" : [
         "SG_ID"
       ],
       "IpProtocol" : [
         "tcp"
       ],
       "FromPort" : [
         80
       ],
   ```

```
    "ToPort" : [
      80
    ],
    "Source" : [
      "10.0.0.1/24"
    ]
  }
}
```

3. Output the RFC template JSON file to a file named RevokeSGIngressRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RevokeSGIngressRfc.json
```

4. Modify and save the RevokeSGIngressRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-1vjbacfr4ufdv",
  "ChangeTypeVersion": "3.0",
  "Title": "Revoke security group ingress rule"
}
```

5. Create the RFC, specifying the RevokeSGIngressRfc file and the RevokeSGIngressParams file:

```
aws amscm create-rfc --cli-input-json file://RevokeSGIngressRfc.json  --execution-
parameters file://RevokeSGIngressParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 3.0.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1vjbacfr4ufdv.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-RevokeSecurityGroupIngressRuleV3",
  "Region" : "us-east-1",
  "Parameters" : {
    "SecurityGroupId" : [
      "sg-abcd1234"
    ],
    "IpProtocol" : [
      "tcp"
    ],
    "FromPort" : [
      "80"
    ],
    "ToPort" : [
      "80"
    ],
    "Source" : [
      "10.0.0.1/32"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RevokeSecurityGroupIngressRuleV3",
  "Region": "us-east-1",
  "Parameters": {
    "SecurityGroupId": [
      "sg-abcd1234"
    ],
    "IpProtocol": [
      "tcp"
    ],
    "FromPort": [
      "80"
    ],
    "ToPort": [
      "80"
    ],
    "Source": [
```

```
      "10.0.0.0"
    ]
  }
}
```

Security Group | Update (Review Required)

Update the inbound and the outbound rules of a security group, and optionally associate it with AWS resources.

Full classification: Management | Advanced stack components | Security group | Update (review required)

Change Type Details

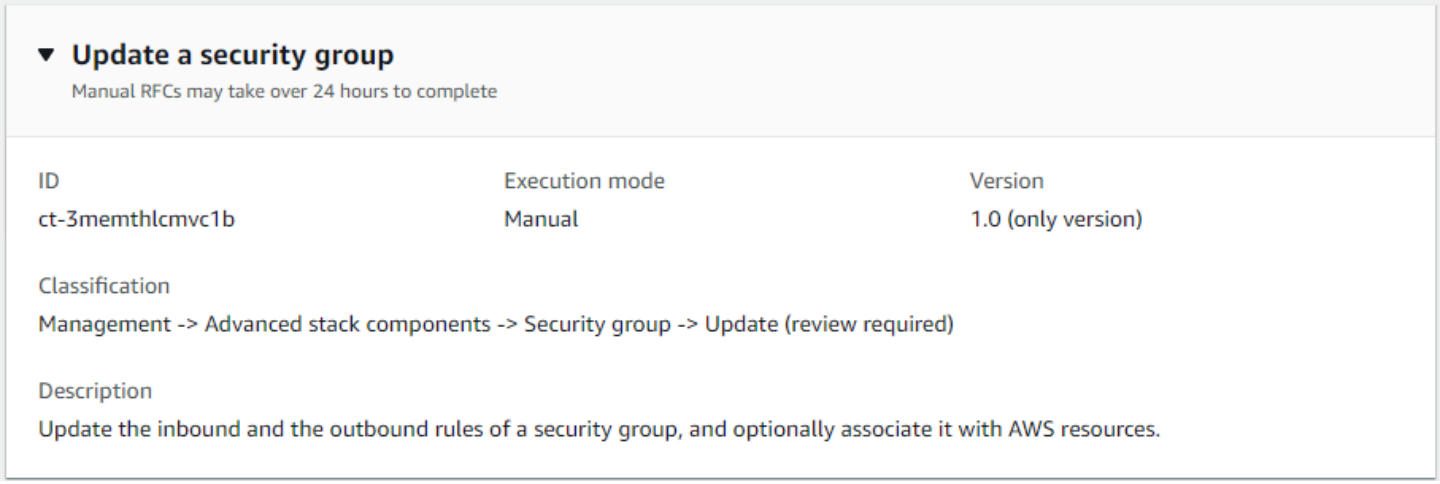
Change type ID	ct-3memthlcmvc1b
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update security group (review required)

Updating an AMS Security Group (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AMS Security Group (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

1. Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

(Optional) To add inbound or outbound rules, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b" --change-
type-version "1.0" --title "Add-SG-Rules" --execution-parameters "{\"SecurityGroupId
\":\"SG_ID\", \"AddInboundRules\":{\"Protocol\":\"TCP\", \"PortRange\":
\"49152-65535\, \"Source\":\"203.0.113.5/32\"}, \"AddOutboundRules\":{\"Protocol\":
\"TCP\", \"PortRange\":\"49152-65535\, \"Destination\":\"203.0.113.5/32\"}}"
```

(Optional) To remove inbound or outbound rules, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b" --
change-type-version "1.0" --title "Remove-SG-Rules" --execution-parameters
 "{\"SecurityGroupId\":\"SG_ID\", \"Name\":\"MA-Test-SG-QC\", \"RemoveInboundRules\":
{\"Protocol\":\"TCP\", \"PortRange\":\"49152-65535\, \"Source\":\"203.0.113.5/32\"},
 \"RemoveOutboundRules\":{\"Protocol\":\"TCP\", \"PortRange\":\"49152-65535\,
 \"Destination\":\"203.0.113.5/32\"}}"
```

(Optional) To add associated resources, you can issue a command similar to this that uses the Update Security Group CT:

```
aws --profile saml amscm create-rfc --change-type-id "ct-3memthlcmvc1b" --
change-type-version "1.0" --title "Add-SG-Resources" --execution-parameters
 "{\"SecurityGroupId\":\"SG_ID\", \"AssociatedResources\":\"IDS_OF_RESOURCES\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateSGParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3memthlcmvc1b" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateSGParams.json
   ```

2. Modify and save the UpdateSGParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "SecurityGroupId": "sg-1234abcd",
  "DisassociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-567890abcdefg1234"
  ]
}
```

3. Output the RFC template JSON file to a file named UpdateSGRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateSGRfc.json
```

4. Modify and save the UpdateSGRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3memthlcmvc1b",
"Title":                "SG-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateSG Rfc file and the UpdateSGParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateSGRfc.json  --execution-
parameters file://UpdateSGParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS security groups, see Amazon EC2 Security Groups for Linux Instances and/or Security Groups for Your VPC.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3memthlcmvc1b.

Example: Required Parameters

```
{
  "SecurityGroupId": "sg-1234abcd",
  "AddAssociatedResources": [],
  "AddInboundRules": [],
  "RemoveInboundRules": [],
  "AddOutboundRules": [],
  "RemoveOutboundRules": []
}
```

Example: All Parameters

```
{
  "SecurityGroupId": "sg-1234abcd",
  "AddAssociatedResources": [
    "i-1234abcd",
    "i-234abcd1",
    "i-34abcd12",
    "i-4abcd123",
    "i-abcd1234",
    "i-1234567890abcdefg",
    "i-234567890abcdefg1",
    "i-34567890abcdefg12",
    "i-4567890abcdefg123",
    "i-567890abcdefg1234"
  ],
  "AddInboundRules": [
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
```

```
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
```

```
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" },
```

```
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16",
 "Description": "Client1" }
  ],
  "RemoveInboundRules": [
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
    { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
```

```
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" },
   { "Protocol": "TCP(6)", "PortRange":"80", "Source": "192.168.0.0/16" }
 ],
 "AddOutboundRules": [
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
   { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
"Description": "Client1" },
```

```
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
```

```
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16",
 "Description": "Client1" }
  ],
  "RemoveOutboundRules": [
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
    { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
```

```
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" },
     { "Protocol": "ALL", "PortRange": "ALL", "Destination": "192.168.0.0/16" }
   ],
   "Priority": "Medium",
   "Tags": [
     { "Key":
  "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
  "Value":
  "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmno
     { "Key": "B", "Value": "bb" },
     { "Key": "C", "Value": "cc" },
     { "Key": "D", "Value": "dd" },
     { "Key": "E", "Value": "ee" },
     { "Key": "F", "Value": "ff" },
```

```
    { "Key": "G", "Value": "gg" },
    { "Key": "H", "Value": "hh" },
    { "Key": "I", "Value": "ii" },
    { "Key": "J", "Value": "jj" },
    { "Key": "K", "Value": "kk" },
    { "Key": "L", "Value": "ll" },
    { "Key": "M", "Value": "mm" },
    { "Key": "N", "Value": "nn" },
    { "Key": "O", "Value": "oo" },
    { "Key": "P", "Value": "pp" },
    { "Key": "Q", "Value": "qq" },
    { "Key": "R", "Value": "rr" },
    { "Key": "S", "Value": "ss" },
    { "Key": "T", "Value": "tt" },
    { "Key": "U", "Value": "uu" },
    { "Key": "V", "Value": "vv" },
    { "Key": "W", "Value": "ww" },
    { "Key": "X", "Value": "xx" },
    { "Key": "Y", "Value": "yy" },
    { "Key": "Z", "Value": "zz" },
    { "Key": "a", "Value": "aa" },
    { "Key": "b", "Value": "bb" },
    { "Key": "c", "Value": "cc" },
    { "Key": "d", "Value": "dd" },
    { "Key": "e", "Value": "ee" },
    { "Key": "f", "Value": "ff" },
    { "Key": "g", "Value": "gg" },
    { "Key": "h", "Value": "hh" },
    { "Key": "i", "Value": "ii" },
    { "Key": "j", "Value": "jj" },
    { "Key": "k", "Value": "kk" },
    { "Key": "l", "Value": "ll" },
    { "Key": "m", "Value": "mm" },
    { "Key": "n", "Value": "nn" },
    { "Key": "o", "Value": "oo" },
    { "Key": "p", "Value": "pp" },
    { "Key": "q", "Value": "qq" },
    { "Key": "r", "Value": "rr" },
    { "Key": "s", "Value": "ss" },
    { "Key": "t", "Value": "tt" },
    { "Key": "u", "Value": "uu" },
    { "Key": "v", "Value": "vv" },
    { "Key": "w", "Value": "ww" },
    { "Key": "x", "Value": "xx" }
```

```
  ]
}
```

Stack | Delete

Delete an existing stack and its resources from your account. The effects of deleting a resource vary. For details, see the appropriate AWS documentation for the resource. Note that termination protection on a resource in the stack causes the RFC to fail. To check for a resource's termination protection status, see the corresponding AWS console.

Full classification: Management | Advanced stack components | Stack | Delete

Change Type Details

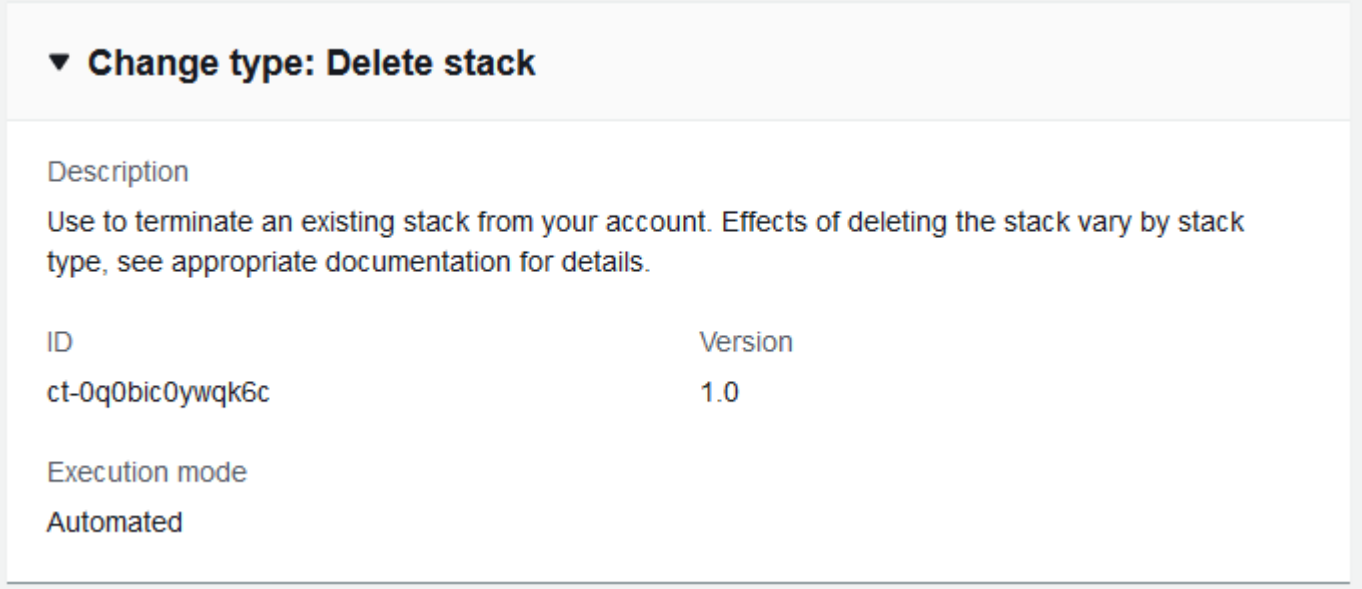
Change type ID	ct-0q0bic0ywqk6c
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete stack

Deleting a Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting a Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0" --
title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder; this example names it DeleteStackRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Modify and save the DeleteStackRfc.json file.

   The internal quotation marks in the ExecutionParameters JSON extension must be escaped with a backslash (\). Example without start and end time:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   "ExecutionParameters":  "{
           \"StackId\":\"STACK_ID\"}"
   }
   ```

3. Create the RFC:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
   ```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If deleting an S3 bucket, it must be emptied of objects first.

> **Important**
>
> Deleting stacks can have unwanted and unanticipated consequences. For important caveats, see RFC Troubleshooting section RFCs for Delete Stack.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0q0bic0ywqk6c.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "StackId": "stack-a1b2c3d4e5f67890e",
  "TimeoutInMinutes": 720
}
```

Stack Patching Configuration | Update

Use to update patch configuration.

Full classification: Management | Advanced stack components | Stack patching configuration | Update

Change Type Details

Change type ID	ct-34alumbtv2b9p
Current version	1.0
Expected execution duration	15 minutes
AWS approval	Required
Customer approval	Not required

Execution mode	Automated

Additional Information

> **Important**
>
> This change type has been deprecated and cannot be used.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-34alumbtv2b9p.

Example: Required Parameters

```
{
  "StackId": "stack-12345678901234567"
}
```

Example: All Parameters

```
{
  "StackId": "stack-12345678901234567",
  "MaintenanceWindow": {
    "DayOfWeek": 4,
    "DurationInMinutes": 240,
    "Hour": 18,
    "Minute": 0,
    "WeekOfMonth": 2
  },
  "HealthyHostThreshold": 0.8
}
```

Tag | Bulk Update

Bulk add tags to existing, supported resources: Autoscaling, EC2, Elastic Load Balancing, RDS and S3 buckets. AMS infrastructure stacks (stacks named mc-*) cannot have tags added with this change type. Use this with AWS Tag Editor when managing large numbers of tags (i.e. >50).

Full classification: Management | Advanced stack components | Tag | Bulk update

Change Type Details

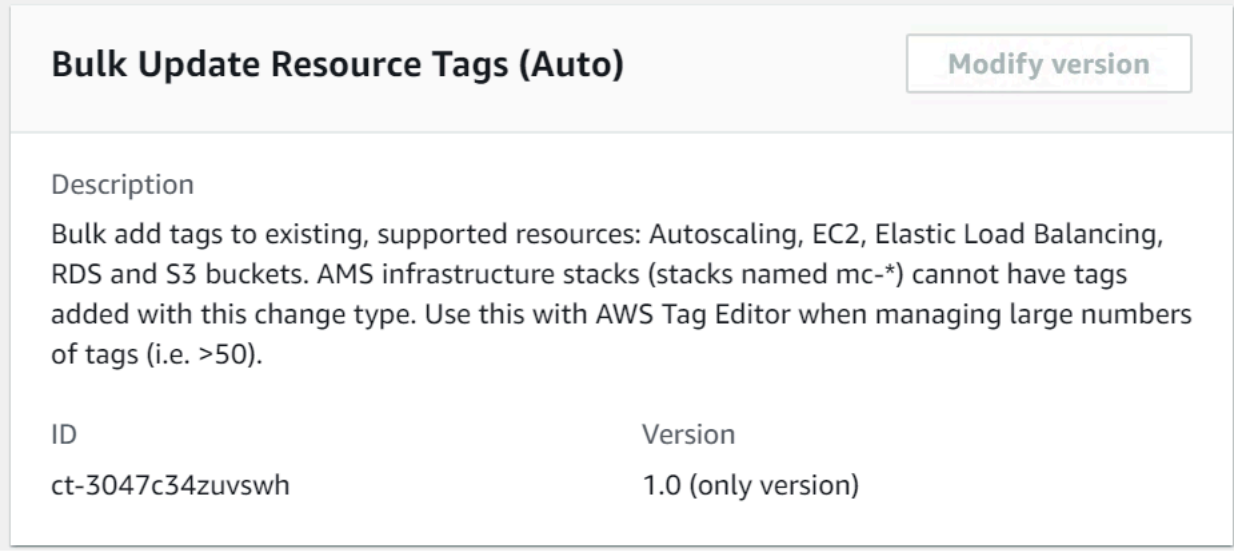
Change type ID	ct-3047c34zuvswh
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Bulk update tags

Bulk Updating Tags with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Bulk Updating Tags with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3047c34zuvswh" --change-type-version "1.0" --
title "Bulk update Tags" --execution-parameters '{"DocumentName":"AWSManagedServices-
BulkUpdateTags","Region":"us-east-1","Parameters":{"CsvS3Url":["PRESIGNED_S3_URL"]}}'
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder. This example names it TagBulkUpdateAutoRfc.json. Note that since there is only one execution parameter for starting a stack, the execution parameter can be in the schema JSON file itself and there is no need to create a separate execution parameters JSON file.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagBulkUpdateAutoRfc.json
   ```

2. Modify and save the TagBulkUpdateAutoRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
    "DocumentName": "AWSManagedServices-BulkUpdateTags",
    "Region": "us-east-1",
    "Parameters": {
        "CsvS3Url": [
          "PRESIGNED_S3_URL"
          ]
    }
}
```

3. Output the RFC template JSON file to a file; this example names it TagBulkUpdateAutoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TagBulkUpdateAutoRfc.json
```

4. Modify and save the TagBulkUpdateAutoRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3047c34zuvswh",
  "Title": "Bulk update Tags"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagBulkUpdateAutoRfc.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- The Tag Editor export populates a matrix of all tags against all resources, missing tags are populated with a value of 'not tagged'. Re-using this export CSV as input to the RFC results in all the previously missing tags being created, with literal values of 'not tagged'.
- This change type is automated, so it typically runs more quickly than a review required change type; however, if your situation is unusual, you might want to use the review required change type for additional help. See Tag | Bulk Update (Review Required).
- For supported services and other information, see Tag bulk update notes.

- Bulk add tags to existing supported resources, except those in AMS infrastructure stacks (stacks named mc-*).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3047c34zuvswh.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-BulkUpdateTags",
    "Region": "us-east-1",
    "Parameters": {
        "CsvS3Url": ["https://example-bucket.s3.amazonaws.com/tags.csv?
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE"]
    }
}
```

Tag | Bulk Update (Review Required)

Bulk add tags to existing, supported resources except those in AMS infrastructure stacks (stacks named mc-*). Tags simplify categorization, identification and targeting AWS resources. Use this with AWS Tag Editor when managing large numbers of tags (i.e. >50). For Autoscaling, EC2, Elastic Load Balancing, RDS resources and S3 buckets, use automated CT ct-3047c34zuvswh.

Full classification: Management | Advanced stack components | Tag | Bulk update (review required)

Change Type Details

Change type ID	ct-0k4b96aatyqgl
Current version	1.0
Expected execution duration	240 minutes

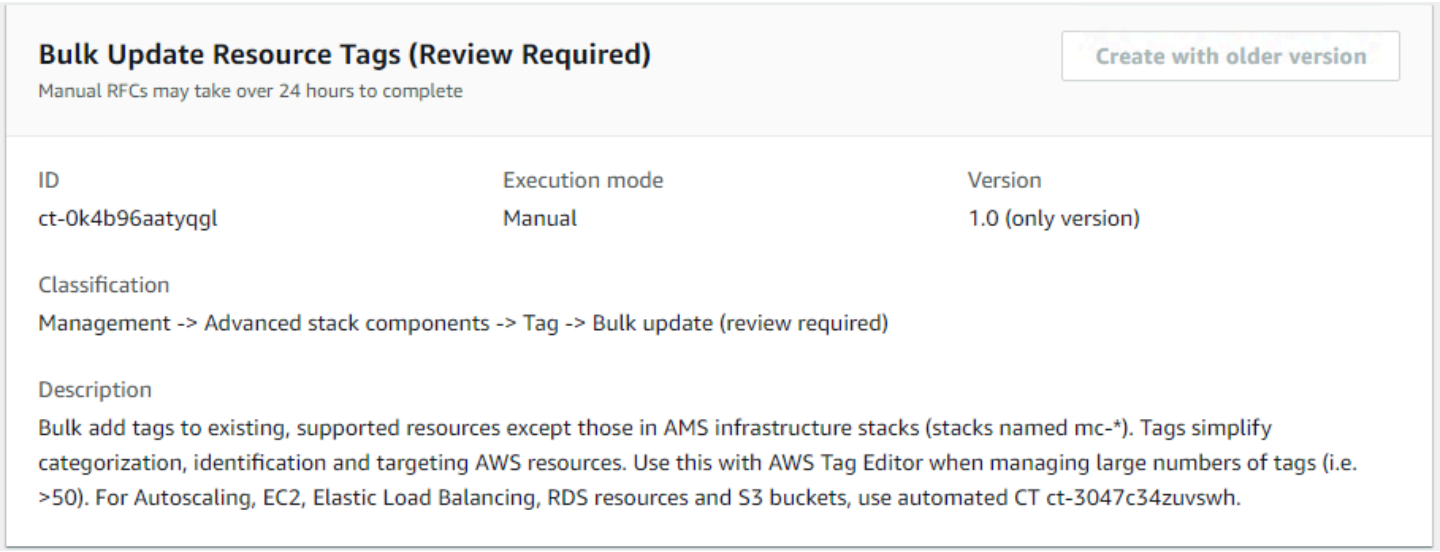
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Bulk update tags (review required)

Bulk Updating Tags (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Bulk Updating Tags (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title bulk-update-tags --change-type-id ct-0k4b96aatyqgl
 --change-type-version 1.0 --execution-parameters '{"Description":"test-tag-bulk-
update","CsvS3Url":"PRE-SIGNED_S3_URL"}'
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder. This example names it TagBulkUpdateRfc.json. Note that since there is only one execution parameter for starting a stack, the execution parameter can be in the schema JSON file itself and there is no need to create a separate execution parameters JSON file.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagBulkUpdateRfc.json
   ```

2. Modify and save the TagBulkUpdateRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-0k4b96aatyqgl",
     "Title": "Bulk-Update_Tags",
     "ExecutionParameters": "{\"Description\":\"Bulk tag resources\",\"CsvS3Url\":
   \"PRESIGNED_S3_URL\"}"
   }}
   ```

3. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagBulkUpdateRfc.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- The Tag Editor export populates a matrix of all tags against all resources, missing tags are populated with a value of 'not tagged'. Re-using this export CSV as input to the RFC results in all the previously missing tags being created, with literal values of 'not tagged'.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

  To use the automated version of this change type, recommended except in unusual circumstances, see Tag | Bulk Update.
- For supported services and other information, see Tag bulk update notes.
- Bulk add tags to existing supported resources, except those in AMS infrastructure stacks (stacks named mc-*).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0k4b96aatyqgl.

Example: Required Parameters

```
{
    "Description": "Tag all the instances for App A",
    "CsvS3Url": "https://example-bucket.s3.eu-central-1.amazonaws.com/tags.csv"
}
```

Example: All Parameters

```
{
  "Description": "Tag all the instances for App A",
  "CsvS3Url": "https://example-bucket.s3.amazonaws.com/tags.csv?
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE",
  "Priority": "Medium"
}
```

Tag | Delete

Delete tags from existing, tagged resources: Autoscaling, EC2, Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally, CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS infrastructure stacks (stacks named mc-*) cannot have tags deleted with this change type.

Full classification: Management | Advanced stack components | Tag | Delete

Change Type Details

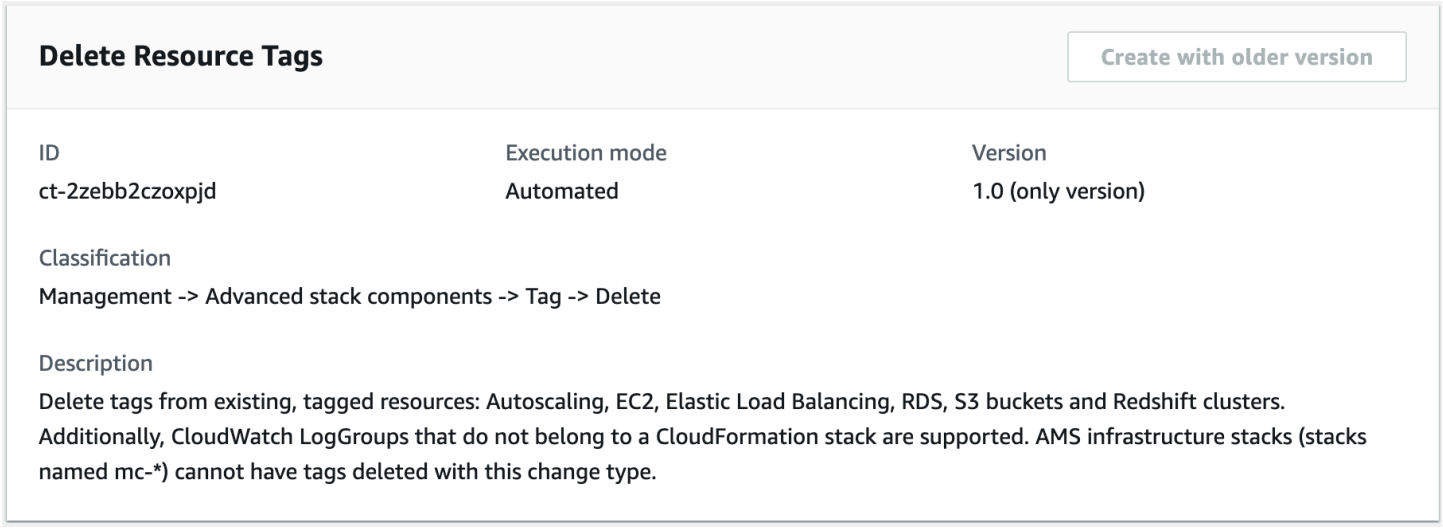
Change type ID	ct-2zebb2czoxpjd
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete tags

Deleting Tags with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting Tags with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2zebb2czoxpjd" --change-type-version "1.0"
 --title "Delete Tags" --execution-parameters '{"DocumentName":"AWSManagedServices-
UpdateTags","Region":"us-east-1","Parameters":{"ResourceArns":
["i-1234567890abcdef0","vol-1234567890abcdef0","arn:aws:rds:us-east-1:123456789012:db/
my-db-instance"],"RemoveTags":["Unused tag 1","Unused tag 2","Unused tag 3"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagDeleteAutoParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagDeleteAutoParams.json
   ```

2. Modify and save the TagDeleteAutoParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateTags",
     "Region": "us-east-1",
     "Parameters":     {
       "ResourceArns": [
         "i-1234567890abcdef0",
         "vol-1234567890abcdef0",
         "arn:aws:rds:us-east-1:123456789012:db/my-db-instance"
       ],
       "RemoveTags": [
         "Unused tag 1",
         "Unused tag 2",
         "Unused tag 3"
       ]
     }
   }
   ```

3. Output the RFC template to a file in your current folder. This example names it TagDeleteAutoRfc.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagDeleteAutoRfc.json
   ```

4. Modify and save the TagDeleteAutoRfc.json file.

   The internal quotation marks in the `ExecutionParameters` JSON extension must be escaped with a backslash (\). Example:

```
{
"ChangeTypeId":         "ct-2zebb2czoxpjd",
"Title":                "Delete-Tags-Auto-RFC"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagDeleteAutoRfc.json  --execution-
parameters file://TagDeleteAutoParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

AMS infrastructure stacks (stacks named `mc-*`) cannot have tags deleted with this change type.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2zebb2czoxpjd.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateTags",
  "Region": "us-east-1",
  "Parameters":     {
    "ResourceArns": [
      "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "arn:aws:ec2:us-east-1:123456789012:volume/vol-1234567890abcdef0",
      "snap-1234567890abcdef0",
```

```
      "arn:aws:rds:us-east-1:123456789012:db/my-db-instance",
      "arn:aws:redshift:us-east-1:123456789012:cluster:my-cluster",
      "arn:aws:logs:ap-southeast-2:123456789012:log-group:my-log-group:*"
    ],
    "RemoveTags": [
      "k4",
      "k5",
      "aws-migration-project-id"
    ]
  }
}
```

Tag | Delete (Review Required)

Delete tags from existing, supported resources except those in AMS infrastructure stacks (stacks named mc-*). For Autoscaling, EC2, Elastic Load Balancing, RDS resources and S3 buckets, use automated CT ct-2zebb2czoxpjd.

Full classification: Management | Advanced stack components | Tag | Delete (review required)

Change Type Details

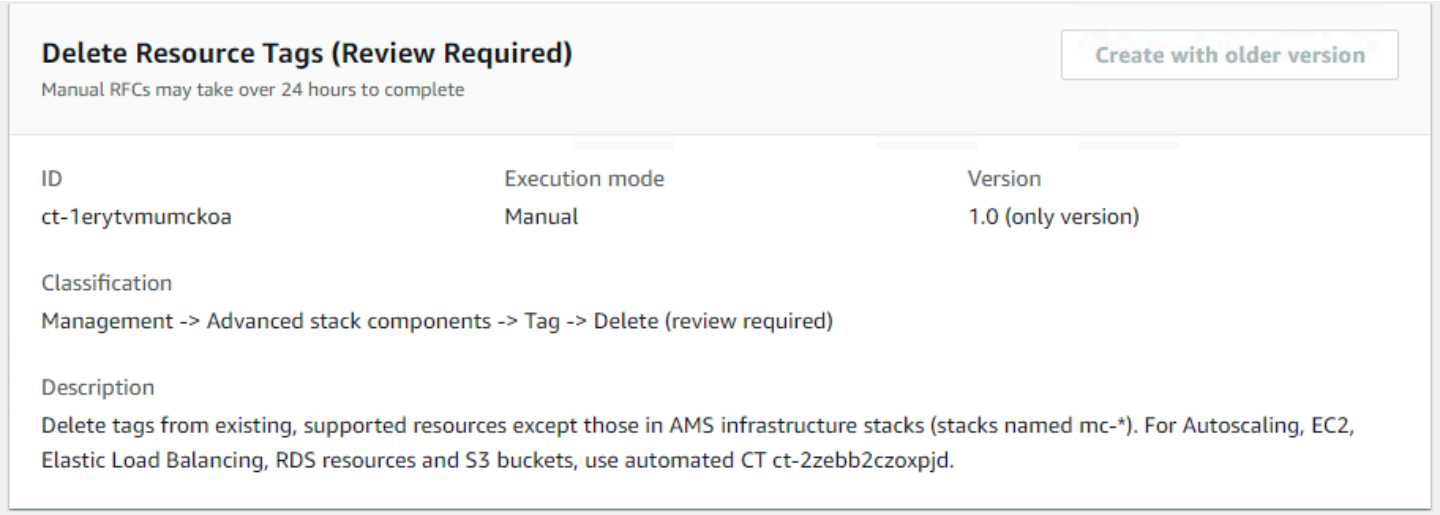
Change type ID	ct-1erytvmumckoa
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete tags (review required)

Deleting Tags (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting Tags (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title delete-tags --change-type-id ct-1erytvmumckoa --
change-type-version 1.0 --execution-parameters '{"Description":"test","Resources":
[{"ResourceArn":"i-abcd1234","RemoveTags":["Name","Owner"]},
{"ResourceArn":"arn:aws:ec2:ap-southeast-2:123456789012:instance/
i-019714a96c22f5452","RemoveTags":["Name","Owner"]}]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagDeleteParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagDeleteParams.json
   ```

2. Modify and save the TagDeleteParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "Description": "Delete tags",
     "Resources": [
       {
         "ResourceArn": "i-abcd1234",
         "RemoveTags": [
           "Unused tag 1",
           "Unused tag 2"
         ]
       },
       {
         "ResourceArn": "arn:aws:ec2:ap-southeast-2:123456789012:instance/
   i-1234567890abcdef1",
         "RemoveTags": [
           "Unused tag 1",
           "Unused tag 2"
         ]
       }
     ]
   }
   ```

3. Output the RFC template to a file in your current folder. This example names it TagDeleteRfc.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagDeleteRfc.json
   ```

4. Modify and save the TagDeleteRfc.json file.

   The internal quotation marks in the `ExecutionParameters` JSON extension must be escaped with a backslash (\). Example:

   ```
   {
   "ChangeTypeId":         "ct-1erytvmumckoa",
   "Title":                "Delete-Tags-RFC"
   }
   ```

5. Create the RFC:

   ```
   aws amscm create-rfc --cli-input-json file://TagDeleteRfc.json  --execution-
   parameters file://TagDeleteParams.json
   ```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1erytvmumckoa.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "Remove tags from instances",
  "Resources": [
```

```
    {
      "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "RemoveTags": ["k1", "k2"]
    },
    {
      "ResourceArn": "i-0fedcba0987654321",
      "RemoveTags": ["k1", "k2"]
    }
  ],
  "Priority": "Medium"
}
```

Tag | Update

Update tags on existing, tagged resources: Autoscaling, EC2, Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally, CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS infrastructure stacks (stacks named mc-*) cannot have tags updated with this change type.

Full classification: Management | Advanced stack components | Tag | Update

Change Type Details

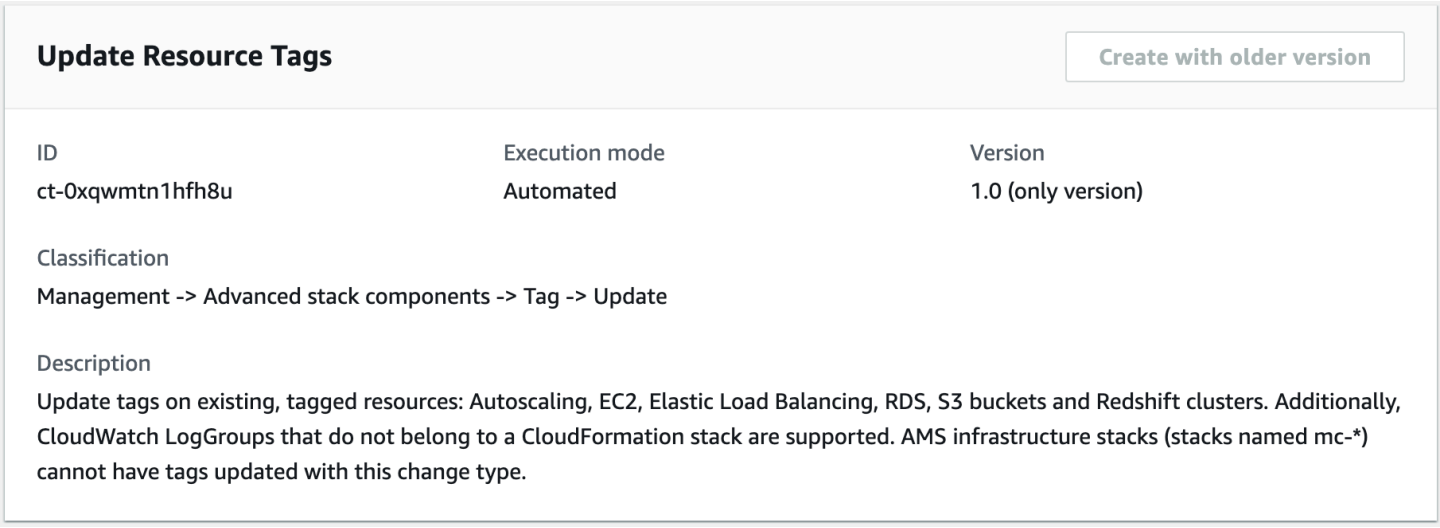
Change type ID	ct-0xqwmtn1hfh8u
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update tags

Updating tags with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating tags with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0xqwmtn1hfh8u" --change-type-version "1.0"
 --title "Update Tags" --execution-parameters '{"DocumentName":"AWSManagedServices-
UpdateTags","Region":"us-east-1","Parameters":{"ResourceArns":
["i-1234567890abcdef0","vol-1234567890abcdef0","arn:aws:rds:us-east-1:123456789012:db/
my-db-instance"],"AddOrUpdateTags":["{\"Key\":\"Name\",\"Value\":\"App1\"}","{\"Key\":
\"Owner\",\"Value\":\"Dev\"}"],"RemoveTags":["Unused tag 1","Unused tag 2","Unused tag
 3"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagUpdateAutoParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagUpdateAutoParams.json
   ```

2. Modify and save the TagUpdateAutoParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateTags",
     "Region": "us-east-1",
     "Parameters":     {
       "ResourceArns": [
         "i-1234567890abcdef0",
         "vol-1234567890abcdef0",
         "arn:aws:rds:us-east-1:123456789012:db/my-db-instance"
       ],
       "AddOrUpdateTags": [
         {\"Key\":\"Name\",\"Value\":\"App1\"},
         {\"Key\":\"Owner\",\"Value\":\"Dev\"}
       ],
       "RemoveTags": [
         "Unused tag 1",
         "Unused tag 2",
         "Unused tag 3"
       ]
     }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it TagUpdateAutoRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TagUpdateAutoRfc.json
```

4. Modify and save the TagUpdateAutoRfc.json file.

   The internal quotation marks in the ExecutionParameters JSON extension must be escaped with a backslash (\). Example without start and end time:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0xqwmtn1hfh8u",
"Title":                "Update-Tags-Auto-RFC"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagUpdateAutoRfc.json  --execution-
parameters file://TagUpdateAutoParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

AMS infrastructure stacks (stacks named `mc-*`) can't have tags updated with this change type. Use the **Bulk update** change type (ct-3047c34zuvswh) if you have more than fifty tags to manage.

Tags can't contain the following prefixes:

- ams
- AMS
- Ams

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0xqwmtn1hfh8u.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateTags",
  "Region": "us-east-1",
  "Parameters":     {
    "ResourceArns": [
      "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "arn:aws:ec2:us-east-1:123456789012:volume/vol-1234567890abcdef0",
      "snap-1234567890abcdef0",
      "arn:aws:rds:us-east-1:123456789012:db/my-db-instance",
      "arn:aws:redshift:us-east-1:123456789012:cluster:my-cluster",
      "arn:aws:logs:ap-southeast-2:123456789012:log-group:my-log-group:*"
    ],
    "AddOrUpdateTags": [
      "{\"Key\":\"k1\",\"Value\":\"v1\"}",
      "{\"Key\":\"k2\",\"Value\":\"v2\"}",
      "{\"Key\":\"aws-migration-project-id\",\"Value\":\"project-id\"}"
    ],
    "RemoveTags": [
      "k4",
      "k5",
      "k6"
    ]
  }
}
```

Tag | Update (Review Required)

Add tags to, update tags on, or remove tags from, existing, supported, resources except those in AMS infrastructure stacks (stacks named mc-*). Tags simplify categorization, identification and targeting AWS resources. Use BulkUpdate if you have >50 tags to manage. For Autoscaling, EC2, Elastic Load Balancing, RDS resources and S3 buckets, use automated CT ct-0xqwmtn1hfh8u.

Full classification: Management | Advanced stack components | Tag | Update (review required)

Change Type Details

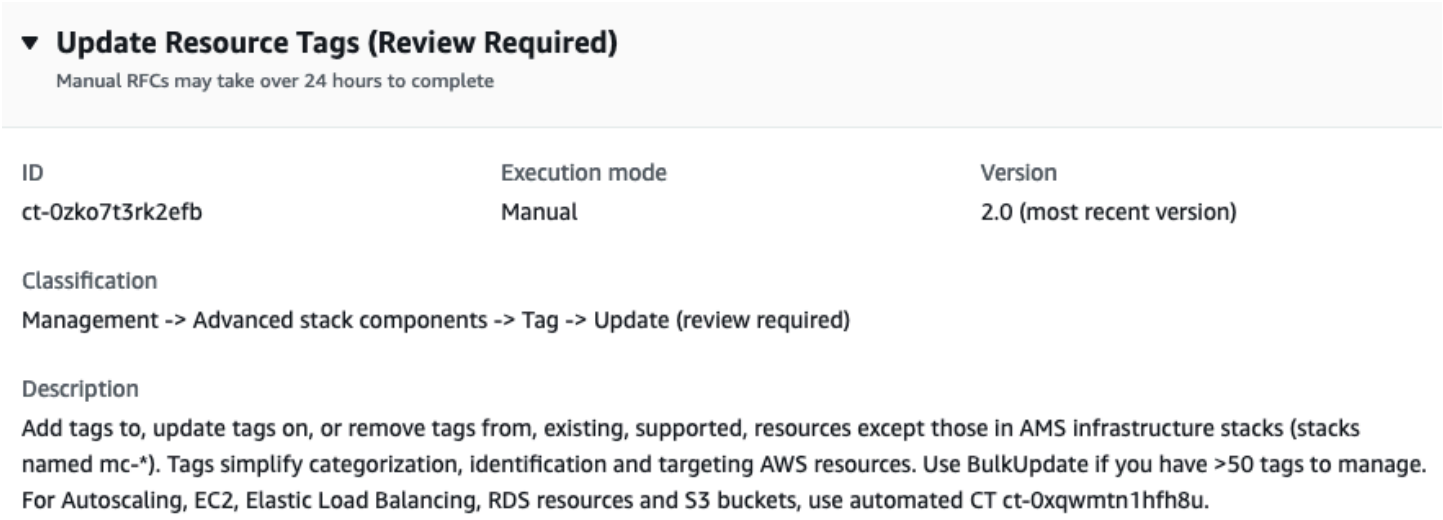
Change type ID	ct-0zko7t3rk2efb
Current version	2.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update tags (review required)

Updating tags (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating tags (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title update-tags --change-type-id ct-0zko7t3rk2efb
 --change-type-version 2.0 --execution-parameters '{"Resources":
[{"ResourceArn":"i-abcd1234","AddOrUpdateTags":[{"Key":"Name","Value":"app-
instance-1"},{"Key":"Owner","Value":"Dep A"}],"RemoveTags":["Team","Prod"]},
{"ResourceArn":"arn:aws:ec2:ap-southeast-2:123456789012:instance/
i-019714a96c22f5452","AddOrUpdateTags":[{"Key":"Name","Value":"app-instance-1"},
{"Key":"Owner","Value":"Dep A"}],"RemoveTags":["Team","Prod"]}]}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema to a file in your current folder. This example names it TagUpdateParams.json.

   ```
   aws amscm create-rfc --generate-cli-skeleton > TagUpdateParams.json
   ```

2. Modify and save the TagUpdateParams.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
    "Resources": [
    {
      "ResourceArn": "i-abcd1234",
      "AddOrUpdateTags": [
        {
          "Key": "Name",
          "Value": "app-instance-1"
        },
        {
          "Key": "Owner",
          "Value": "Dep A"
        }
      ],
      "RemoveTags": [
        "Unused tag 1",
        "Unused tag 2"
      ]
    },
    {
      "ResourceArn": "arn:aws:ec2:ap-southeast-2:123456789012:instance/
i-1234567890abcdef1",
      "AddOrUpdateTags": [
        {
          "Key": "Name",
          "Value": "app-instance-1"
        },
        {
          "Key": "Owner",
          "Value": "Dep A"
        }
      ],
      "RemoveTags": [
        "Unused tag 1",
        "Unused tag 2"
      ]
    }
  ]
}
```

3. Output the RFC template to a file in your current folder; this example names it TagUpdateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TagUpdateRfc.json
```

4. Modify and save the TagUpdateRfc.json file.

   The internal quotation marks in the ExecutionParameters JSON extension must be escaped with a backslash (\). Example without start and end time:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-0zko7t3rk2efb",
"Title":                "Update-Tags-RFC"
}
```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://TagUpdateRfc.json  --execution-
parameters file://TagUpdateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0zko7t3rk2efb.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Resources": [
    {
```

```
      "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0",
      "AddOrUpdateTags": [
        {
          "Key": "k1",
          "Value": "v1"
        },
        {
          "Key": "k2",
          "Value": "v2"
        },
        {
          "Key": "k3",
          "Value": "v3"
        }
      ],
      "RemoveTags": ["k4", "k5", "k6"]
    },
    {
      "ResourceArn": "i-0fedcba0987654321",
      "AddOrUpdateTags": [
        {
          "Key": "k1",
          "Value": "v1"
        },
        {
          "Key": "k2",
          "Value": "v2"
        },
        {
          "Key": "k3",
          "Value": "v3"
        }
      ],
      "RemoveTags": ["k4", "k5", "k6"]
    }
  ],
  "Priority": "Medium"
}
```

Target Group | Attach Instances

Attach instance or instances to the target group (ALB and NLB).

Full classification: Management | Advanced stack components | Target group | Attach instances

Change Type Details

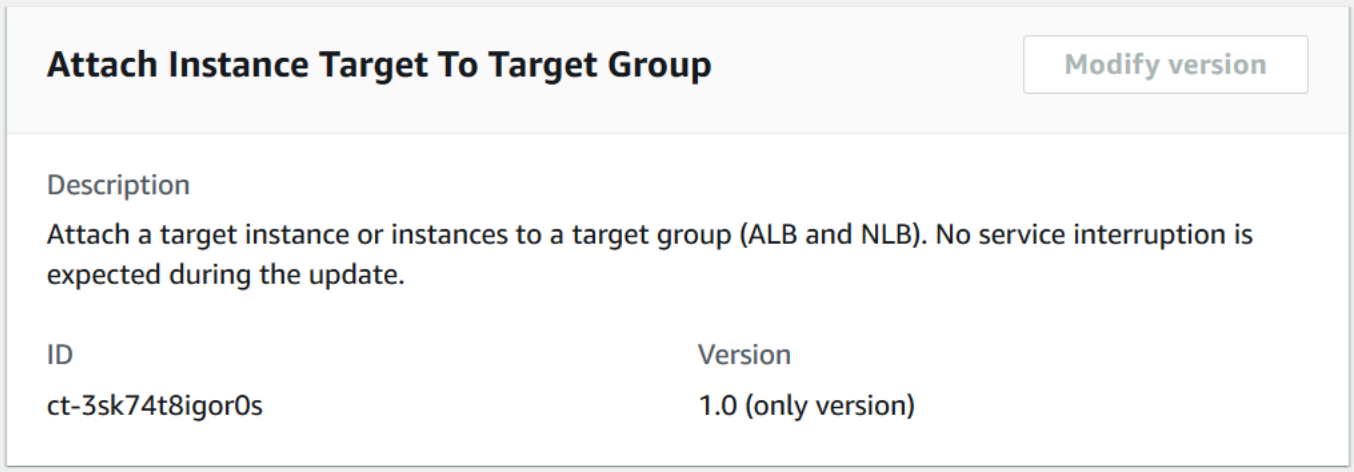
Change type ID	ct-3sk74t8igor0s
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Attach instances to a target group

Attaching Instances to a Target Group with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Attaching Instances to a Target Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-3sk74t8igor0s" \
--change-type-version "1.0" --title "AttachInstancesToTargetGroup" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
AttachInstancesToTargetGroup\",\"Region\":\"us-east-1\",\"Parameters\":{\"InstancesIds
\":[\"i-000000000000\",\"i-111111111111\"],\"InstancesPort\":[\"80\"],\"TargetGroupArn
\":[\"arn:aws:elasticloadbalancing:us-east-1:00000000000:targetgroup/test-target-
group/0000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it TgAttachInstanceParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3sk74t8igor0s"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TgAttachInstanceParams.json
   ```

2. Modify and save the TgAttachInstanceParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-AttachInstancesToTargetGroup",
  "Region": "us-east-1",
  "Parameters": {
      "InstancesIds": [
          "i-000000000000",
          "i-111111111111"
      ],
      "InstancesPort": [
          "80"
      ],
      "TargetGroupArn": [
          "arn:aws:elasticloadbalancing:us-east-1:00000000000:targetgroup/test-
target-group/0000000000"
      ]
  }
}
```

3. Output the RFC template to a file in your current folder named TgAttachInstanceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TgAttachInstanceRfc.json
```

4. Modify and save the TgAttachInstanceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3sk74t8igor0s",
"Title":                "Target-Group-Attach-Instance-RFC"
}
```

5. Create the RFC, specifying the TgAttachInstanceRfc file and the TgAttachInstanceParams file:

```
aws amscm create-rfc --cli-input-json file://TgAttachInstanceRfc.json --execution-
parameters file://TgAttachInstanceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about target groups, see ELB Target Groups.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3sk74t8igor0s.

Example: Required Parameters

```
 {
  "DocumentName": "AWSManagedServices-AttachInstancesToTargetGroup",
  "Region": "us-east-1",
  "Parameters": {
    "InstancesIds": ["i-0000000000"],
    "InstancesPort": ["80"],
    "TargetGroupArn": ["arn:aws:elasticloadbalancing:eu-
west-1:000000000000:targetgroup/target-group-name/000000000000"]
  }
}
```

Example: All Parameters

```
 {
  "DocumentName": "AWSManagedServices-AttachInstancesToTargetGroup",
  "Region": "us-east-1",
  "Parameters": {
    "InstancesIds": ["i-0000000000"],
    "InstancesPort": ["80"],
    "TargetGroupArn": ["arn:aws:elasticloadbalancing:eu-
west-1:000000000000:targetgroup/target-group-name/000000000000"]
  }
}
```

Target Group | Detach Instances

Detach an instance, or instances, from the specified port of a target group (ALB or NLB).

Full classification: Management | Advanced stack components | Target group | Detach instances

Change Type Details

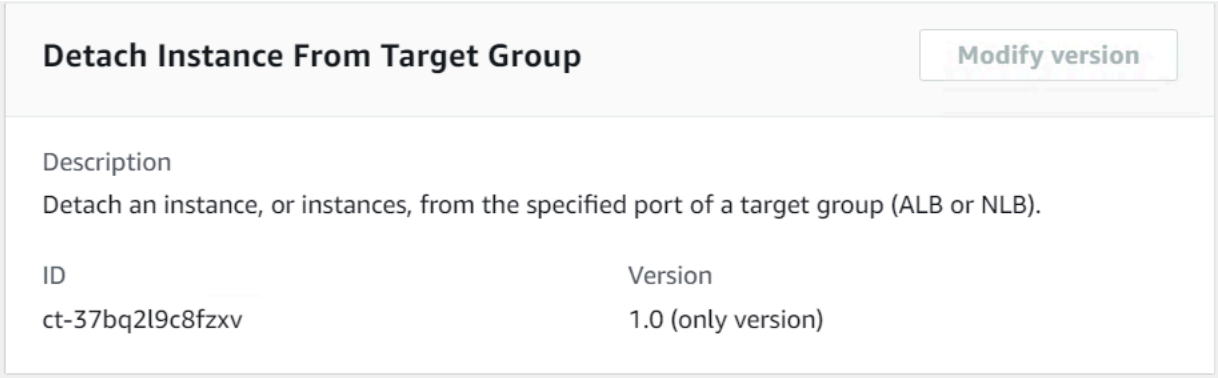
Change type ID	ct-37bq2l9c8fzxv
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Detach instances from a target group

Detaching Instances from a Target Group with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Detaching Instances from a Target Group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-37bq2l9c8fzxv" \
--change-type-version "1.0" --title "DetachInstancesFromTargetGroup" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
DetachInstancesFromTargetGroup\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstancesIds\":[\"i-000000000000\",\"i-111111111111\"],\"InstancesPort\":[\"80\"],
\"TargetGroupArn\":[\"arn:aws:elasticloadbalancing:us-east-1:00000000000:targetgroup/
test-target-group/0000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it TgDetachInstanceParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-37bq2l9c8fzxv"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TgDetachInstanceParams.json
   ```

2. Modify and save the TgDetachInstanceParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DetachInstancesFromTargetGroup",
     "Region": "us-east-1",
   ```

```
  "Parameters": {
      "InstancesIds": [
          "i-000000000000",
          "i-111111111111"
      ],
      "InstancesPort": [
          "80"
      ],
      "TargetGroupArn": [
          "arn:aws:elasticloadbalancing:us-east-1:00000000000:targetgroup/test-
target-group/0000000000"
      ]
  }
}
```

3. Output the RFC template to a file in your current folder named TgDetachInstanceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TgDetachInstanceRfc.json
```

4. Modify and save the TgDetachInstanceRfc.json file. For example, you can replace the contents with something like this:

   Version 1.0:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-37bq2l9c8fzxv",
"Title":                "Target-Group-Detach-Instance-RFC"
}
```

5. Create the RFC, specifying the TgDetachInstanceRfc file and the TgDetachInstanceParams file:

```
aws amscm create-rfc --cli-input-json file://TgDetachInstanceRfc.json --execution-
parameters file://TgDetachInstanceParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about target groups, see ELB Target Groups.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-37bq2l9c8fzxv.

Example: Required Parameters

```
 {
  "DocumentName": "AWSManagedServices-DetachInstancesFromTargetGroup",
  "Region": "us-east-1",
  "Parameters": {
    "InstancesIds": ["i-0000000000"],
    "InstancesPort": ["80"],
    "TargetGroupArn": ["arn:aws:elasticloadbalancing:eu-
west-1:000000000000:targetgroup/target-group-name/000000000000"]
  }
}
```

Example: All Parameters

```
 {
  "DocumentName": "AWSManagedServices-DetachInstancesFromTargetGroup",
  "Region": "us-east-1",
  "Parameters": {
    "InstancesIds": ["i-0000000000"],
    "InstancesPort": ["80"],
    "TargetGroupArn": ["arn:aws:elasticloadbalancing:eu-
west-1:000000000000:targetgroup/target-group-name/000000000000"]
  }
}
```

Target Group | Update (For ALB)

Use to update properties of an existing Target Group for an Application Load Balancer created by CT id ct-1r19m51jeijlk.

Full classification: Management | Advanced stack components | Target group | Update (for ALB)

Change Type Details

Change type ID	ct-2v82sp4np40ki

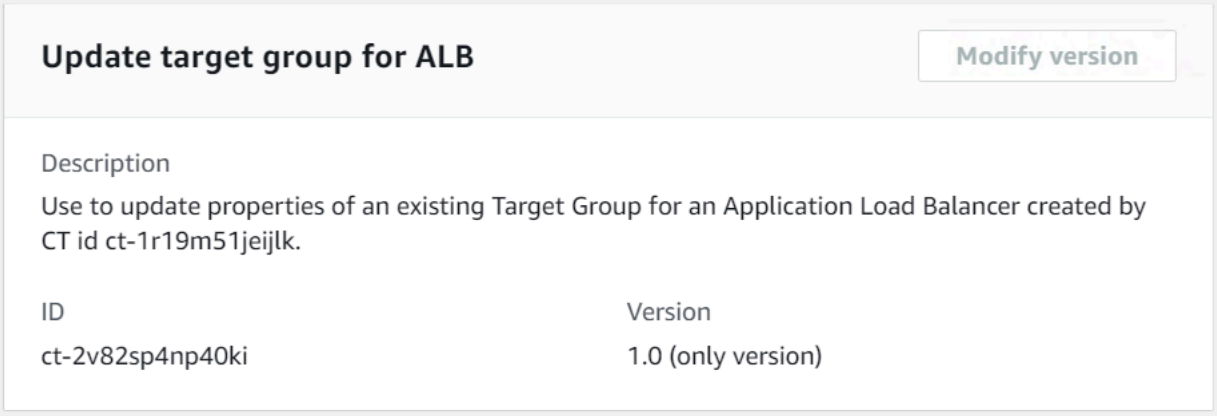
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update ALB target group

Updating a Target Group for an Application Load Balancer with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a Target Group for an Application Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title update-tg-alb --change-type-id ct-2v82sp4np40ki --change-
type-version 1.0 --execution-parameters '{"Description":"Update target group for
 ALB","VpcId":"vpc-1234abcd","StackTemplateId":"stm-9c1t8maqho0os5k22","Name":"update-
tg-alb","TimeoutInMinutes":60,"Parameters":{"HealthCheckHealthyThreshold":
 "5","HealthCheckUnhealthyThreshold":
 "3","HealthCheckInterval": 30,"HealthCheckTimeout": "10","HealthCheckTargetPath":
 "/healthcheck","HealthCheckTargetPort": "80","HealthCheckTargetProtocol":
 "HTTP","ValidHTTPCode": "200-259","DeregistrationDelayTimeout":
 "300","SlowStartDuration": "60","StickinessCookieExpirationPeriod":
 "3600","Target1ID": "i-abcdef01","Target1Port": "80","Target1AvailabilityZone":
 "AZ","Target2ID": "i-abcdefabcdefabcd1","Target2Port": "80","Target2AvailabilityZone":
 "AZ"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it CreateTgAlbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2v82sp4np40ki" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateTgAlbParams.json
   ```

2. Modify and save the UpdateTgAlbParams file. For example, you can replace the contents with something like this:

```
{
"Description":      "Target-Group-ALB-Create",
"VpcId":            "VPC_ID",
"Name":             "My-ALB-Target-Group",

"Parameters":   {
    "LoadBalancerArn":              ARN,
    "DefaultActionTargetGroupArn":  ARN,
    "Port":                         PORT,
    "Protocol":                    Protocol"
    }
}
```

3. Output the RFC template to a file in your current folder named UpdateTgAlbRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateTgAlbRfc.json
```

4. Modify and save the UpdateTgAlbRfc.json file. For example, you can replace the contents with something like this:

   Version 1.0:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2v82sp4np40ki",
"Title":                "Target-Group-ALB-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateTgAlbRfc file and the UpdateTgAlbParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateTgAlbRfc.json --execution-
parameters file://UpdateTgAlbParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> The 2.0 version of this change type uses a different StackTemplateId (stm-9c1t8maqho0os5k22) from the 1.0 version. This is important if you are submitting the RFC with this change type version at the command line. The new version includes a new, required, parameter: **ApplicationLoadBalancer**.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2v82sp4np40ki.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-123456789abcdefgh",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-123456789abcdefgh",
  "Parameters": {
    "HealthCheckHealthyThreshold": "5",
    "HealthCheckUnhealthyThreshold": "3",
    "HealthCheckInterval": 30,
    "HealthCheckTimeout": "10",
    "HealthCheckTargetPath": "/healthcheck",
    "HealthCheckTargetPort": "80",
    "HealthCheckTargetProtocol": "HTTP",
    "ValidHTTPCode": "200-259",
    "DeregistrationDelayTimeout": "300",
    "SlowStartDuration": "60",
    "StickinessCookieExpirationPeriod": "3600",
    "Target1ID": "i-abcdef01",
```

```
    "Target1AvailabilityZone": "",
    "Target2ID": "i-abcdefabcdefabcd1",
    "Target2AvailabilityZone": "",
    "Target3ID": "i-abcdefabcdefabcd2",
    "Target3AvailabilityZone": "",
    "Target4ID": "i-abcdefabcdefabcd3",
    "Target4AvailabilityZone": "",
    "Target5ID": "i-abcdefabcdefabcd4",
    "Target5AvailabilityZone": "",
    "Target6ID": "i-abcdefabcdefabcd5",
    "Target6AvailabilityZone": "",
    "Target7ID": "i-abcdefabcdefabcd6",
    "Target7AvailabilityZone": "",
    "Target8ID": "i-abcdefabcdefabcd7",
    "Target8AvailabilityZone": ""
  }
}
```

Target Group | Update (For NLB)

Use to update properties of an existing Target Group for a Network Load Balancer.

Full classification: Management | Advanced stack components | Target group | Update (for NLB)

Change Type Details

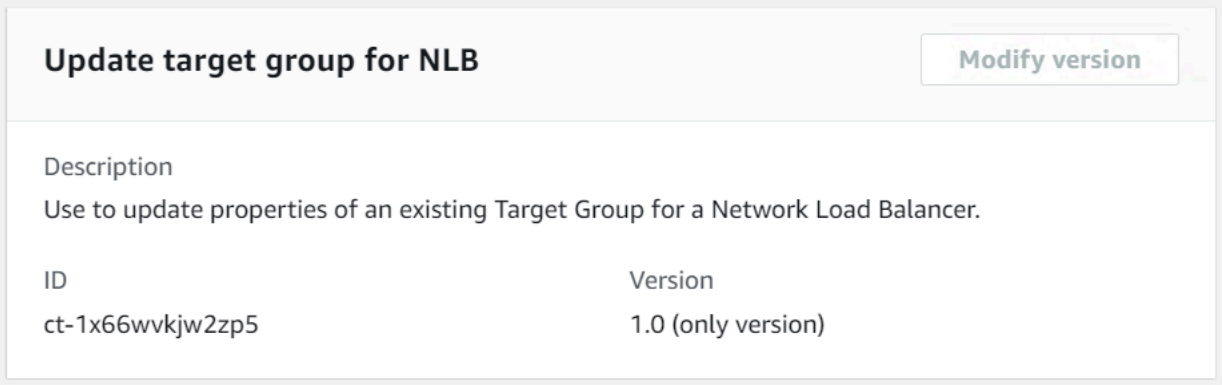
Change type ID	ct-1x66wvkjw2zp5
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update NLB target group

Updating a Target Group for a Network Load Balancer with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a Target Group for a Network Load Balancer with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title update-tg-nlb --change-type-id ct-1x66wvkjw2zp5 --change-
type-version 1.0 --execution-parameters '{"Description":"Update target group for
 NLB","VpcId":"vpc-1234abcd","StackTemplateId":"stm-6pvp2f7cp481g1r47","Name":"test-
update-tg-nlb","TimeoutInMinutes":60,"Parameters":{"HealthCheckHealthyThreshold":
 "5","HealthCheckInterval": 30,"HealthCheckTargetPath": "/
healthcheck","HealthCheckTargetPort": "80","HealthCheckTargetProtocol":
 "HTTP","ProxyProtocolV2": "true","DeregistrationDelayTimeout": "300","Target1ID":
 "i-abcdef01","Target1Port": "80","Target1AvailabilityZone": "AZ","Target2ID": "i-
abcdefabcdefabcd1","Target2Port": "80","Target2AvailabilityZone": "AZ"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateTgNlbParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1x66wvkjw2zp5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateTgNlbParams.json
   ```

2. Modify and save the UpdateTgNlbParams file. For example, you can replace the contents with something like this:

   ```
   {
   "VpcId":            "VPC_ID",
   "StackId":          "STACK_ID",
   "Parameters":   {
       "DeregistrationDelayTimeout":  160
       }
   }
   ```

3. Output the RFC template to a file in your current folder named UpdateTgNlbRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateTgNlbRfc.json
   ```

4. Modify and save the UpdateTgNlbRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1x66wvkjw2zp5",
"Title":                "Target-Group-NLB-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateTgNlbRfc file and the UpdateTgNlbParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateTgNlbRfc.json --execution-
parameters file://UpdateTgNlbParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about target groups, see ELB Target Groups.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1x66wvkjw2zp5.

Example: Required Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-123456789abcdefgh",
  "Parameters": {}
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-1234567890abcdef0",
  "StackId": "stack-123456789abcdefgh",
  "Parameters": {
    "HealthCheckHealthyThreshold": "5",
    "HealthCheckInterval": 30,
    "HealthCheckTargetPath": "/healthcheck",
    "HealthCheckTargetPort": "80",
```

```
    "HealthCheckTargetProtocol": "HTTP",
    "ProxyProtocolV2": "true",
    "DeregistrationDelayTimeout": "300",
    "Target1ID": "192.168.0.1",
    "Target1Port": "80",
    "Target1AvailabilityZone": "all",
    "Target2ID": "192.168.0.2",
    "Target2Port": "80",
    "Target2AvailabilityZone": "all",
    "Target3ID": "10.44.4.125",
    "Target3Port": "8080",
    "Target3AvailabilityZone": "",
    "Target4ID": "10.44.4.126",
    "Target4Port": "8080",
    "Target4AvailabilityZone": "",
    "Target5ID": "192.168.0.127",
    "Target5Port": "80",
    "Target5AvailabilityZone": "all",
    "Target6ID": "192.168.0.128",
    "Target6Port": "80",
    "Target6AvailabilityZone": "all",
    "Target7ID": "192.168.0.129",
    "Target7Port": "8080",
    "Target7AvailabilityZone": "",
    "Target8ID": "192.168.0.130",
    "Target8Port": "8080",
    "Target8AvailabilityZone": ""
  }
}
```

AMS Resource Scheduler Subcategory

Change Type Items and Operations in the AMS Resource Scheduler Subcategory

- Period | Add
- Period | Delete
- Period | Describe
- Period | Update
- Schedule | Add
- Schedule | Delete
- Schedule | Describe

- Schedule | Update
- Solution | Update
- State | Disable
- State | Enable

Period | Add

Add a new period to use with AMS Resource Scheduler. Periods are used in schedules to precisely define when a resource should run.

Full classification: Management | AMS Resource Scheduler | Period | Add

Change Type Details

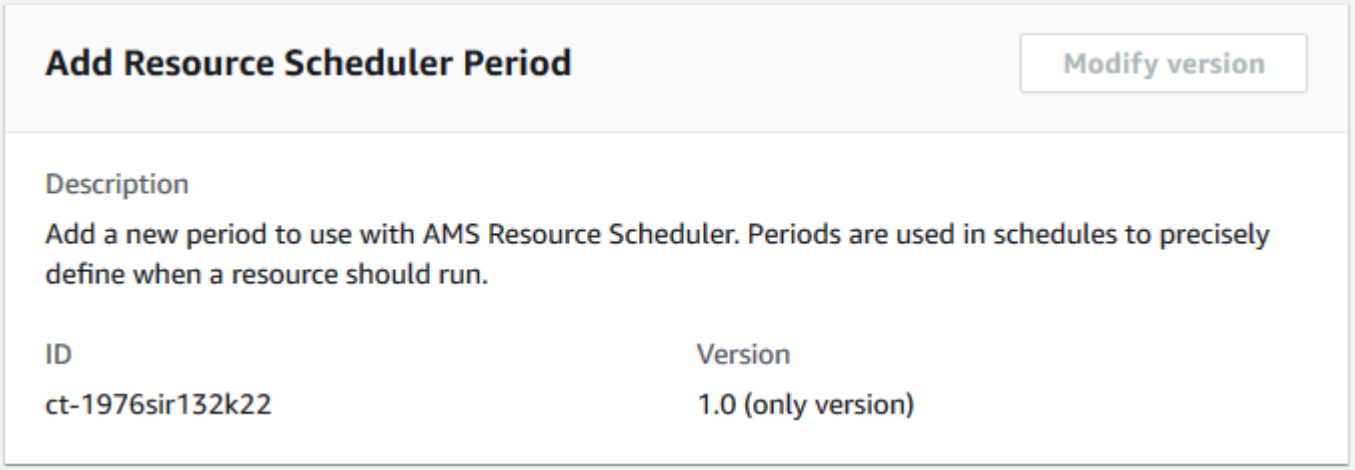
Change type ID	ct-1976sir132k22
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Period add

Adding an AMS Resource Scheduler period with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AMS Resource Scheduler period with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1976sir132k22" --change-type-version "1.0" --
title "Add period for AMS Resource Scheduler" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-AddOrUpdatePeriod\",\"Region\":\"us-east-1\",\"Parameters
\":{\"Action\":[\"add\"],\"Name\":[\"period01\"],\"Description\":[\"Test period
 definition\"],\"BeginTime\":[\"09:00\"],\"EndTime\":[\"17:00\"],\"Months\":[\"jan-
feb\"],\"MonthDays\":[\"jan/3\"],\"WeekDays\":[\"mon-fri\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it AddPeriodParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1976sir132k22" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddPeriodParams.json
   ```

2. Modify and save the AddPeriodParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
     "Region" : "us-east-1",
     "Parameters" : {
       "Action" : ["add"],
       "Name" : ["period01"],
       "Description" : ["Test period definition"],
       "BeginTime" : ["09:00"],
       "EndTime" : ["17:00"],
       "Months" : ["jan-feb"],
       "MonthDays" : ["jan/3"],
       "WeekDays" : ["mon-fri"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it AddPeriodRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AddPeriodRfc.json
   ```

4. Modify and save the AddPeriodRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1976sir132k22",
"Title":                "Add period for AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the AddPeriodRfc file and the AddPeriodParams file:

```
aws amscm create-rfc --cli-input-json file://AddPeriodRfc.json --execution-
parameters file://AddPeriodParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1976sir132k22.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["add"],
    "Name" : ["period01"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
```

```
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["add"],
    "Name" : ["period01"],
    "Description" : ["Test period definition"],
    "BeginTime" : ["09:00"],
    "EndTime" : ["17:00"],
    "Months" : ["jan-feb"],
    "MonthDays" : ["jan/3"],
    "WeekDays" : ["mon-fri"]
  }
}
```

Period | Delete

Delete an existing period used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Period | Delete

Change Type Details

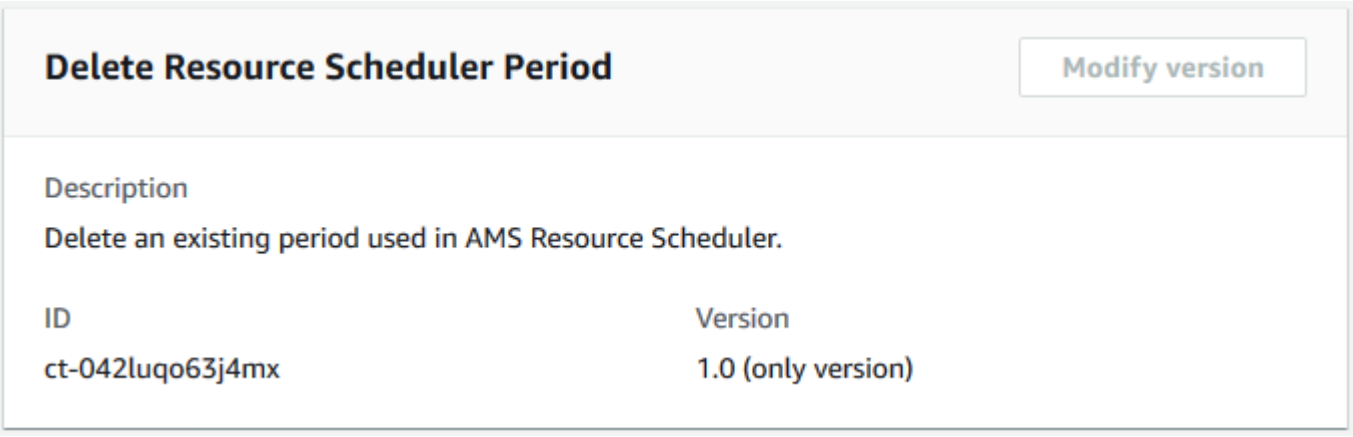
Change type ID	ct-042luqo63j4mx
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Period delete

Deleting an AMS Resource Scheduler Period with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AMS Resource Scheduler period with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-042luqo63j4mx" --change-type-version
 "1.0" --title "Delete period used in AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-DeleteScheduleOrPeriod\",\"Region\":\"us-
east-1\",\"Parameters\":{\"ConfigurationType\":[\"period\"],\"Name\":[\"period01\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DeletePeriodParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-042luqo63j4mx" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeletePeriodParams.json
   ```

2. Modify and save the DeletePeriodParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-DeleteScheduleOrPeriod",
     "Region" : "us-east-1",
     "Parameters" : {
       "ConfigurationType" : ["period"],
       "Name" : ["period01"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it DeletePeriodRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeletePeriodRfc.json
   ```

4. Modify and save the DeletePeriodRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-042luqo63j4mx",
   "Title":                "Delete period used in AMS Resource Scheduler"
   }
   ```

5. Create the RFC, specifying the DeletePeriodRfc file and the DeletePeriodParams file:

```
aws amscm create-rfc --cli-input-json file://DeletePeriodRfc.json --execution-
parameters file://DeletePeriodParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-042luqo63j4mx.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteScheduleOrPeriod",
  "Region" : "us-east-1",
  "Parameters" : {
    "ConfigurationType" : ["period"],
    "Name" : ["period01"]
  }
}
```

Period | Describe

Describe existing periods used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Period | Describe

Change Type Details

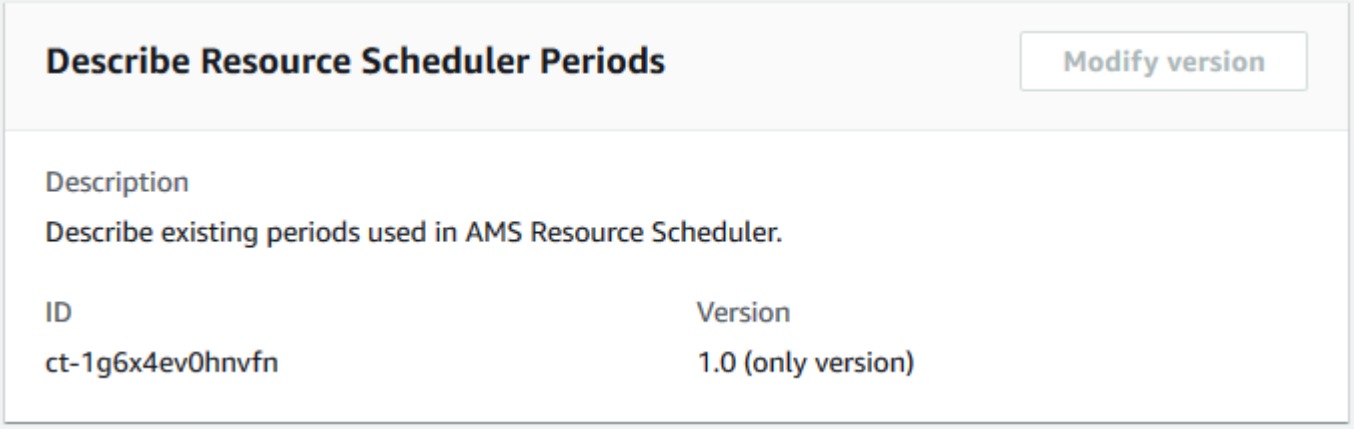
Change type ID	ct-1g6x4ev0hnvfn
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Period describe

Describing an AMS Resource Scheduler period with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Describing an AMS Resource Scheduler period with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1g6x4ev0hnvfn" --change-type-version "1.0"
 --title "Describe periods used in AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-DescribeScheduleOrPeriods\",\"Region\":\"us-
east-1\",\"Parameters\":{\"ConfigurationType\":[\"periods\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DescribePeriodParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1g6x4ev0hnvfn" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DescribePeriodParams.json
   ```

2. Modify and save the DescribePeriodParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-DescribeScheduleOrPeriods",
     "Region" : "us-east-1",
     "Parameters" : {
       "ConfigurationType" : ["periods"]
   ```

```
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it DescribePeriodRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DescribePeriodRfc.json
```

4. Modify and save the DescribePeriodRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1g6x4ev0hnvfn",
"Title":                "Describe periods used in AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the DescribePeriodRfc file and the DescribePeriodParams file:

```
aws amscm create-rfc --cli-input-json file://DescribePeriodRfc.json --execution-
parameters file://DescribePeriodParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1g6x4ev0hnvfn.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DescribeScheduleOrPeriods",
  "Region" : "us-east-1",
  "Parameters" : {
    "ConfigurationType" : ["periods"]
  }
}
```

Period | Update

Update an existing period used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Period | Update

Change Type Details

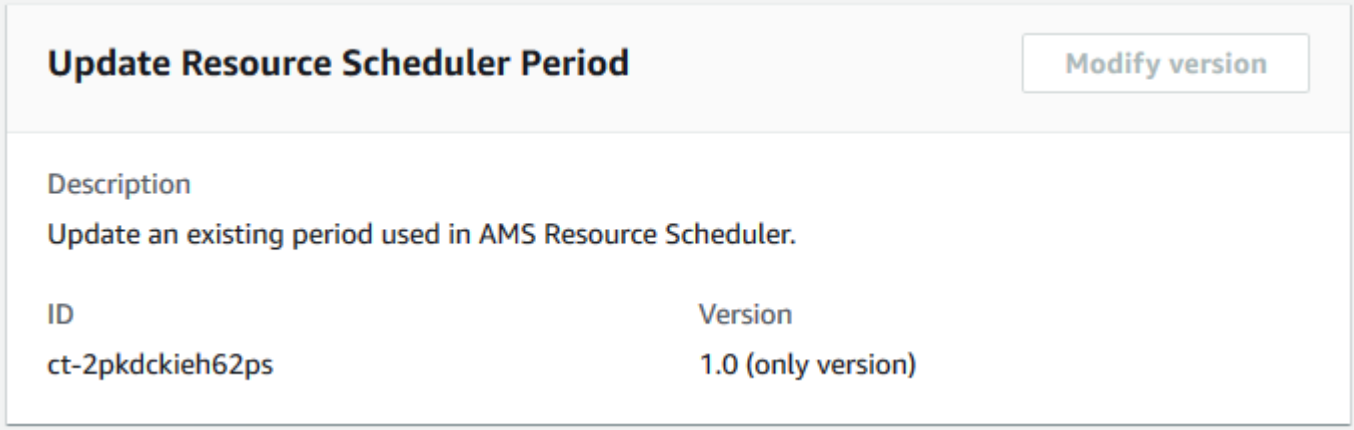
Change type ID	ct-2pkdckieh62ps
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Period update

Updating an AMS Resource Scheduler period with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AMS Resource Scheduler period with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2pkdckieh62ps" --change-type-version
 "1.0" --title "Update period used in AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-AddOrUpdatePeriod\",\"Region\":\"us-east-1\",
\"Parameters\":{\"Action\":[\"update\"],\"Name\":[\"period01\"],\"Description\":[\"Test
 period definition\"],\"BeginTime\":[\"09:00\"],\"EndTime\":[\"17:00\"],\"Months\":
[\"jan-feb\"],\"MonthDays\":[\"jan/3\"],\"WeekDays\":[\"mon-fri\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdatePeriodParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2pkdckieh62ps" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdatePeriodParams.json
   ```

2. Modify and save the UpdatePeriodParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
     "Region" : "us-east-1",
     "Parameters" : {
       "Action" : ["update"],
       "Name" : ["period01"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdatePeriodRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdatePeriodRfc.json
   ```

4. Modify and save the UpdatePeriodRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-2pkdckieh62ps",
   "Title":                "Update period used in AMS Resource Scheduler"
   }
   ```

5. Create the RFC, specifying the UpdatePeriodRfc file and the UpdatePeriodParams file:

```
aws amscm create-rfc --cli-input-json file://UpdatePeriodRfc.json --execution-
parameters file://UpdatePeriodParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2pkdckieh62ps.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["update"],
    "Name" : ["period01"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdatePeriod",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["update"],
    "Name" : ["period01"],
    "Description" : ["Test period definition"],
    "BeginTime" : ["09:00"],
```

```
    "EndTime" : ["17:00"],
    "Months" : ["jan-feb"],
    "MonthDays" : ["jan/3"],
    "WeekDays" : ["mon-fri"]
  }
}
```

Schedule | Add

Add a new schedule to be used in AMS Resource Scheduler. Schedules employ defined periods to determine when the specified resource should run.

Full classification: Management | AMS Resource Scheduler | Schedule | Add

Change Type Details

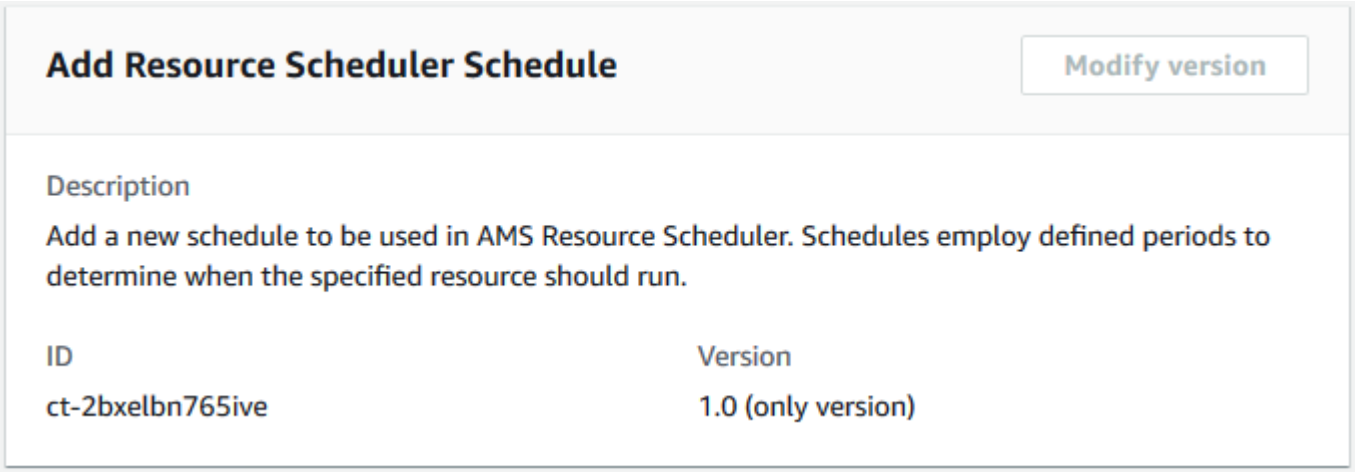
Change type ID	ct-2bxelbn765ive
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Schedule add

Adding an AMS Resource Scheduler schedule with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AMS Resource Scheduler schedule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2bxelbn765ive" --change-type-version
 "1.0" --title "Add a schedule for AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-AddOrUpdateSchedule\",\"Region\":
\"us-east-1\",\"Parameters\":{\"Action\":[\"add\"],\"Name\":[\"Schedule01\"],
\"Description\":[\"Test schedule\"],\"Hibernate\":[\"true\"],\"Enforced\":
[\"false\"],\"OverrideStatus\":[\"running\"],\"Periods\":[\"period01\",\"period02\"],
\"RetainRunning\":[\"false\"],\"StopNewInstances\":[\"true\"],\"SSMMaintenanceWindow\":
[\"window01\"],\"TimeZone\":[\"Australia/Sydney\"],\"UseMaintenanceWindow\":[\"true\"],
\"UseMetrics\":[\"false\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it AddScheduleParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2bxelbn765ive" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddScheduleParams.json
   ```

2. Modify and save the AddScheduleParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
     "Region" : "us-east-1",
     "Parameters" : {
       "Action" : ["add"],
       "Name" : ["Schedule01"],
       "Description" : ["Test schedule"],
       "Hibernate" : ["true"],
       "Enforced" : ["false"],
       "OverrideStatus" : ["running"],
       "Periods" : [
         "period01",
         "period02"
       ],
       "RetainRunning" : ["false"],
       "StopNewInstances" : ["true"],
       "SSMMaintenanceWindow" : ["window01"],
       "TimeZone" : ["Australia/Sydney"],
       "UseMaintenanceWindow" : ["true"],
       "UseMetrics" : ["false"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it AddScheduleRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AddScheduleRfc.json
```

4. Modify and save the AddScheduleRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2bxelbn765ive",
"Title":                "Add a schedule for AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the AddScheduleRfc file and the AddScheduleParams file:

```
aws amscm create-rfc --cli-input-json file://AddScheduleRfc.json --execution-
parameters file://AddScheduleParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2bxelbn765ive.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["add"],
```

```
    "Name" : ["schedule01"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["add"],
    "Name" : ["Schedule01"],
    "Description" : ["Test schedule"],
    "Hibernate" : ["true"],
    "Enforced" : ["false"],
    "OverrideStatus" : ["running"],
    "Periods" : ["period01, period02"],
    "RetainRunning" : ["false"],
    "StopNewInstances" : ["true"],
    "SSMMaintenanceWindow" : ["window01, window02"],
    "TimeZone" : ["Australia/Sydney"],
    "UseMaintenanceWindow" : ["true"],
    "UseMetrics" : ["false"]
  }
}
```

Schedule | Delete

Delete an existing schedule used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Schedule | Delete

Change Type Details

Change type ID	ct-3rk1nl1ufn5g3
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required

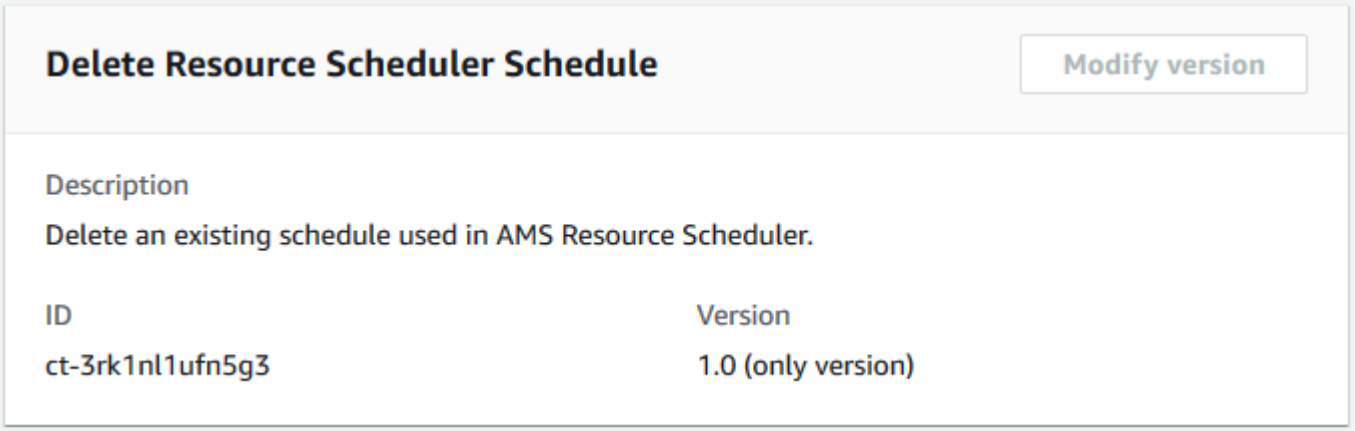
Customer approval	Not required
Execution mode	Automated

Additional Information

Schedule delete

Deleting an AMS Resource Scheduler schedule with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an AMS Resource Scheduler schedule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3rk1nl1ufn5g3" --change-type-version
 "1.0" --title "Delete schedule for AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-DeleteScheduleOrPeriod\",\"Region\":
\"us-east-1\",\"Parameters\":{\"ConfigurationType\":[\"schedule\"],\"Name\":
[\"schedule01\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DeleteScheduleParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3rk1nl1ufn5g3" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteScheduleParams.json
   ```

2. Modify and save the DeleteScheduleParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-DeleteScheduleOrPeriod",
     "Region" : "us-east-1",
     "Parameters" : {
       "ConfigurationType" : ["schedule"],
       "Name" : ["schedule01"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it DeleteScheduleRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteScheduleRfc.json
   ```

4. Modify and save the DeleteScheduleRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3rk1nl1ufn5g3",
"Title":                "Delete schedule for AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the DeleteScheduleRfc file and the DeleteScheduleParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteScheduleRfc.json --execution-
parameters file://DeleteScheduleParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3rk1nl1ufn5g3.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteScheduleOrPeriod",
  "Region" : "us-east-1",
  "Parameters" : {
    "ConfigurationType" : ["schedule"],
    "Name" : ["schedule01"]
```

```
  }
}
```

Schedule | Describe

Describe (generate a detailed list) of existing schedules used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Schedule | Describe

Change Type Details

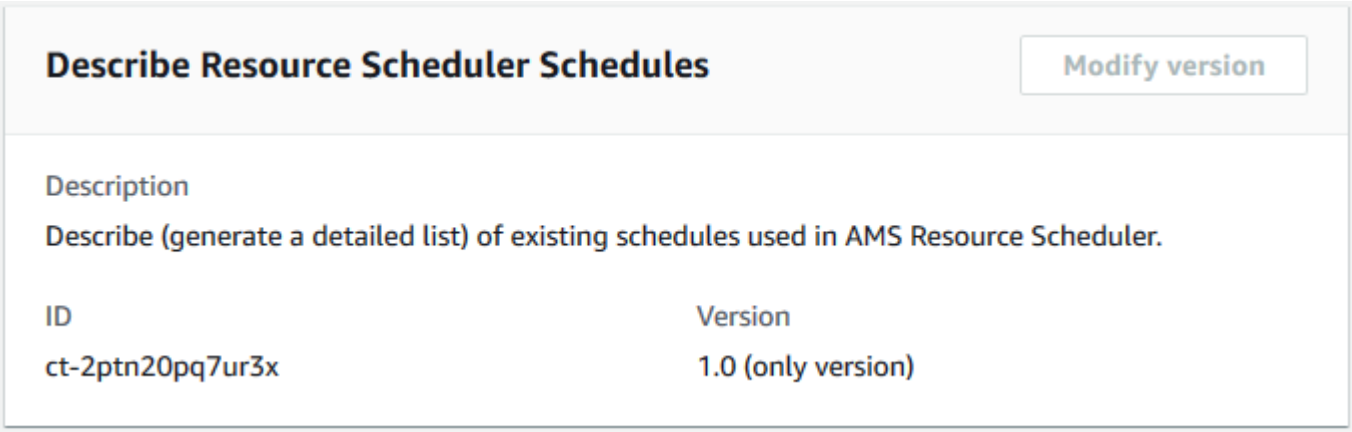
Change type ID	ct-2ptn20pq7ur3x
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Schedule describe

Describing an AMS Resource Scheduler schedule with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Describing an AMS Resource Scheduler schedule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2ptn20pq7ur3x" --change-type-version "1.0"
 --title "Describe schedules used in AMS Resource Scheduler" --execution-parameters
 "{\"DocumentName\":\"AWSManagedServices-DescribeScheduleOrPeriods\",\"Region\":\"us-
east-1\",\"Parameters\":{\"ConfigurationType\":[\"schedules\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DescribeScheduleParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2ptn20pq7ur3x"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DescribeScheduleParams.json
   ```

2. Modify and save the DescribeScheduleParams file.

```
{
  "DocumentName" : "AWSManagedServices-DescribeScheduleOrPeriods",
  "Region" : "us-east-1",
  "Parameters" : {
    "ConfigurationType" : ["schedules"]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it DescribeScheduleRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DescribeScheduleRfc.json
```

4. Modify and save the DescribeScheduleRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2ptn20pq7ur3x",
"Title":                "Describe schedule for AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the DescribeScheduleRfc file and the DescribeScheduleParams file:

```
aws amscm create-rfc --cli-input-json file://DescribeScheduleRfc.json --execution-
parameters file://DescribeScheduleParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2ptn20pq7ur3x.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DescribeScheduleOrPeriods",
  "Region" : "us-east-1",
  "Parameters" : {
    "ConfigurationType" : ["schedules"]
  }
}
```

Schedule | Update

Update an existing schedule to be used in AMS Resource Scheduler.

Full classification: Management | AMS Resource Scheduler | Schedule | Update

Change Type Details

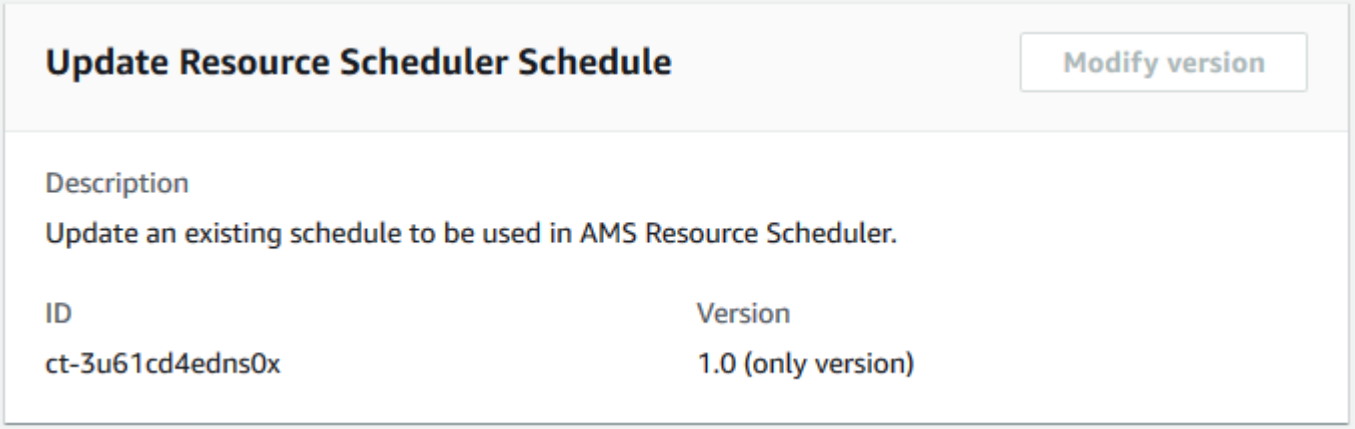
Change type ID	ct-3u61cd4edns0x
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Schedule update

Updating an AMS Resource Scheduler schedule with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AMS Resource Scheduler schedule with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3u61cd4edns0x" --change-type-version
 "1.0" --title "Update a schedule used in AMS Resource Scheduler" --execution-
parameters "{\"DocumentName\":\"AWSManagedServices-AddOrUpdateSchedule\",\"Region
\":\"us-east-1\",\"Parameters\":{\"Action\":[\"update\"],\"Name\":[\"Schedule01\"],
\"Description\":[\"Test schedule\"],\"Hibernate\":[\"true\"],\"Enforced\":
[\"false\"],\"OverrideStatus\":[\"running\"],\"Periods\":[\"period01\",\"period02\"],
\"RetainRunning\":[\"false\"],\"StopNewInstances\":[\"true\"],\"SSMMaintenanceWindow\":
[\"window01\"],\"TimeZone\":[\"Australia/Sydney\"],\"UseMaintenanceWindow\":[\"true\"],
\"UseMetrics\":[\"false\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateScheduleParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3u61cd4edns0x" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateScheduleParams.json
   ```

2. Modify and save the UpdateScheduleParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
     "Region" : "us-east-1",
     "Parameters" : {
       "Action" : ["update"],
       "Name" : ["Schedule01"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it UpdateScheduleRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateScheduleRfc.json
   ```

4. Modify and save the UpdateScheduleRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2ptn3u61cd4edns0x20pq7ur3x",
"Title":                "Update a schedule for AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the UpdateScheduleRfc file and the UpdateScheduleParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateScheduleRfc.json --execution-
parameters file://UpdateScheduleParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3u61cd4edns0x.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["update"],
    "Name" : ["Schedule01"]
  }
}
```

Example: All Parameters

```
{
```

```
  "DocumentName" : "AWSManagedServices-AddOrUpdateSchedule",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["update"],
    "Name" : ["Schedule01"],
    "Description" : ["Test schedule"],
    "Hibernate" : ["true"],
    "Enforced" : ["false"],
    "OverrideStatus" : ["running"],
    "Periods" : ["period01, period02"],
    "RetainRunning" : ["false"],
    "StopNewInstances" : ["true"],
    "SSMMaintenanceWindow" : ["window01, window02"],
    "TimeZone" : ["Australia/Sydney"],
    "UseMaintenanceWindow" : ["true"],
    "UseMetrics" : ["false"]
  }
}
```

Solution | Update

Update the AMS Resource Scheduler solution in the account.

Full classification: Management | AMS Resource Scheduler | Solution | Update

Change Type Details

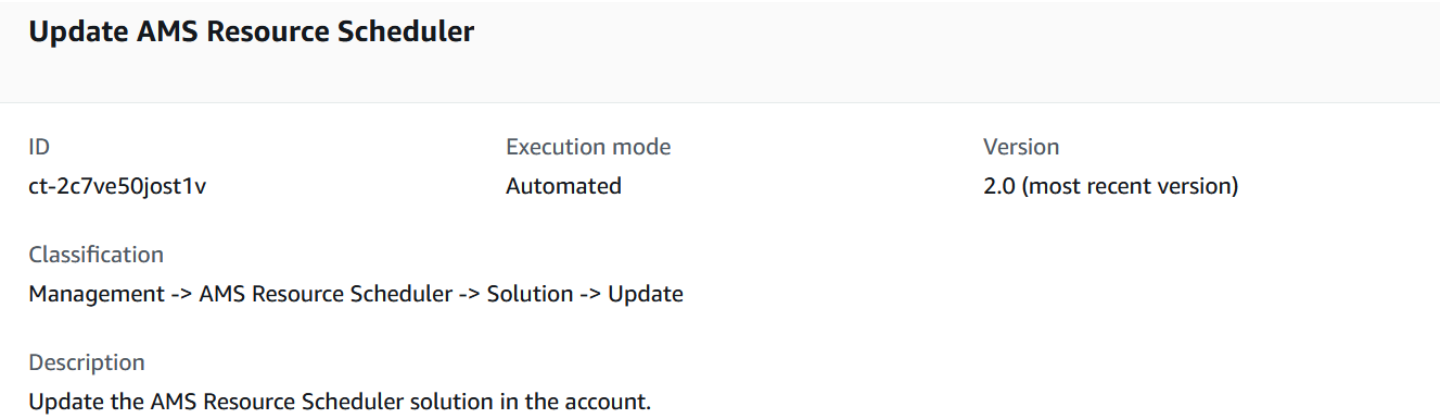
Change type ID	ct-2c7ve50jost1v
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update AMS Resource Scheduler Solution

Updating AMS Resource Scheduler solution with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating AMS Resource Scheduler solution with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id ct-2c7ve50jost1v --change-type-
version "2.0" --title "Update Resource Scheduler Configurations"
 --execution-parameters '{"DocumentName":"AWSManagedServices-
HandleAMSResourceSchedulerStack-Admin","Region":"us-east-1","Parameters":
{"SchedulingActive":["Yes"],"ScheduledServices":["ec2,rds,autoscaling"],"TagName":
["Schedule"],"DefaultTimezone":["America/New_York"],"Action":["Update"]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateResSchedulerParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2c7ve50jost1v"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateResSchedulerParams.json
   ```

2. Modify and save the UpdateResSchedulerParams file.

   ```
   {
     "DocumentName": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "SchedulingActive": [
         "Yes"
       ],
       "ScheduledServices": [
         "ec2,rds,autoscaling"
       ],
       "TagName": [
         "Schedule"
       ],
       "DefaultTimezone": [
         "America/New_York"
       ],
       "Action": [
         "Update"
   ```

```
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateResSchedulerRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateResSchedulerRfc.json
```

4. Modify and save the UpdateResSchedulerRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2c7ve50jost1v",
"Title":                "Update Resource Scheduler Configurations"
}
```

5. Create the RFC, specifying the UpdateResSchedulerRfc file and the UpdateResSchedulerParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateResSchedulerRfc.json --
execution-parameters file://UpdateResSchedulerParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For background information, see How the AMS Resource Scheduler works. For a quick-start tutorial, see AMS Resource Scheduler quick start.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2c7ve50jost1v.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Action" : ["Update"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SchedulingActive" : [
      ""
    ],
    "ScheduledServices" : [
      ""
    ],
    "TagName" : [
      ""
    ],
    "DefaultTimezone" : [
      ""
    ],
    "UseCMK" : [
      "arn:aws:kms:ap-southeast-1:830123456789:key/07aaab3c-50d3-4cd8-
ab61-3de57127dab9"
    ],
    "UseLicenseManager" : [
      "arn:aws:license-manager:ap-southeast-1:830123456789:license-
configuration:lic-78c1e0cfc1233a4eac7197d7ee57f92c"
    ],
    "Action" : [
      "Update"
    ]
  }
}
```

State | Disable

Disable AMS Resource Scheduler in the account. This will prevent resources from being scheduled for automatic start or stop actions even if they are configured for such actions.

Full classification: Management | AMS Resource Scheduler | State | Disable

Change Type Details

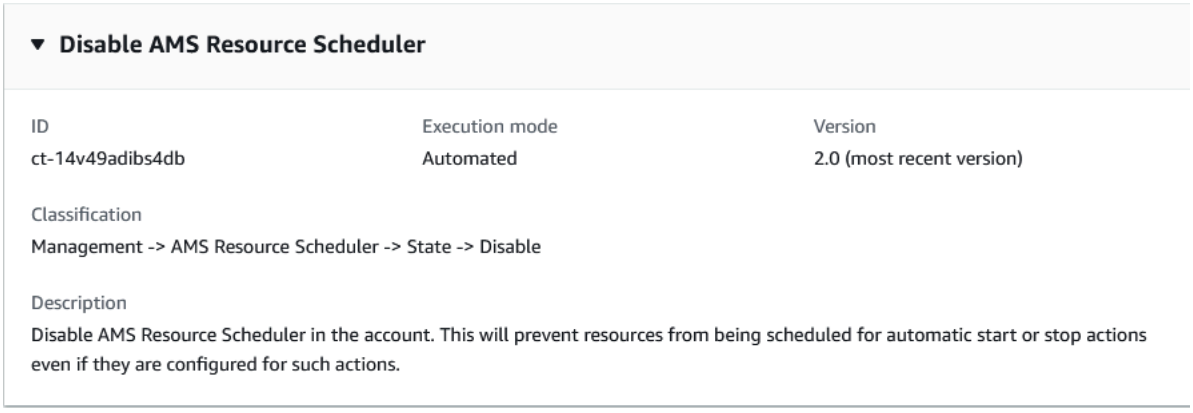
Change type ID	ct-14v49adibs4db
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Disable

Disabling AMS Resource Scheduler with the Console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Disabling AMS Resource Scheduler with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-14v49adibs4db" --change-type-version "2.0"
 --title "Disable AMS Resource Scheduler" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-HandleAMSResourceSchedulerStack-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"SchedulingActive\":[\"No\"],\"Action\":\"Update\"}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DisableResSchedulerParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-14v49adibs4db"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DisableResSchedulerParams.json
   ```

2. Modify and save the DisableResSchedulerParams file.

```
{
  "DocumentName" : "AWSManagedServices-EnableOrDisableAMSResourceScheduler",
  "Region" : "us-east-1",
  "Parameters" : {
  "SchedulingActive" : ["No"],
  "Action" : "Update"
}
}
```

3. Output the RFC template to a file in your current folder; this example names it DisableResSchedulerRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DisableResSchedulerRfc.json
```

4. Modify and save the DisableResSchedulerRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-14v49adibs4db",
"Title":                "Disable AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the DisableResSchedulerRfc file and the DisableResSchedulerParams file:

```
aws amscm create-rfc --cli-input-json file://DisableResSchedulerRfc.json --
execution-parameters file://DisableResSchedulerParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-14v49adibs4db.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "SchedulingActive" : ["No"],
    "Action" : "Update"
  }
}
```

State | Enable

Enable AMS Resource Scheduler in the account where it was previously disabled. This will re-enable scheduling of resources for automatic start or stop actions where the resources are already tagged with a valid schedule. Make sure to verify currently tagged resources and schedules before enabling the scheduler.

Full classification: Management | AMS Resource Scheduler | State | Enable

Change Type Details

Change type ID	ct-2wrvu4kca9xky
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required

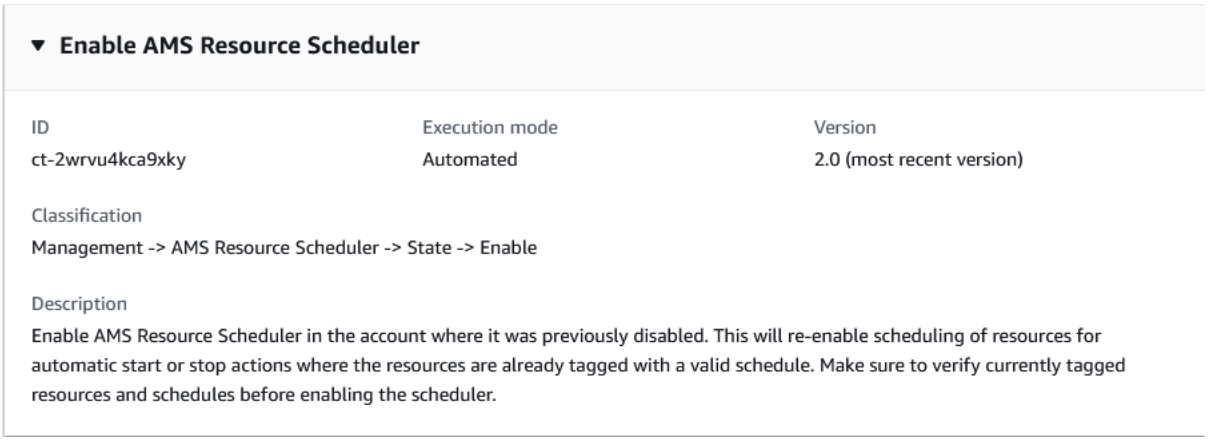
Execution mode	Automated

Additional Information

Enable

Enabling AMS Resource Scheduler with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling AMS Resource Scheduler with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2wrvu4kca9xky" --change-type-version "2.0"
 --title "Enable AMS Resource Scheduler" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-HandleAMSResourceSchedulerStack-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"SchedulingActive\":[\"Yes\"],\"Action\":\"Update\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it EnableResSchedulerParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2wrvu4kca9xky"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EnableResSchedulerParams.json
   ```

2. Modify and save the EnableResSchedulerParams file.

   ```
   {
     "DocumentName" : "AWSManagedServices-EnableOrDisableAMSResourceScheduler",
     "Region" : "us-east-1",
     "Parameters" : {
       "SchedulingActive" : ["Yes"],
       "Action" : "Update"
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it EnableResSchedulerRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableResSchedulerRfc.json
   ```

4. Modify and save the EnableResSchedulerRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2wrvu4kca9xky",
"Title":                "Enable AMS Resource Scheduler"
}
```

5. Create the RFC, specifying the EnableResSchedulerRfc file and the EnableResSchedulerParams file:

```
aws amscm create-rfc --cli-input-json file://EnableResSchedulerRfc.json --
execution-parameters file://EnableResSchedulerParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information, see How the AMS Resource Scheduler works.

AMS Resource Scheduler is based on the AWS Instance Scheduler; to learn more, see AWS Instance Scheduler.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2wrvu4kca9xky.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
```

```
    "SchedulingActive" : ["Yes"],
    "Action" : "Update"
  }
}
```

Applications Subcategory

Change Type Items and Operations in the Applications Subcategory

- IAM Instance Profile | Create (Review Required)

IAM Instance Profile | Create (Review Required)

Use to create an instance profile.

Full classification: Management | Applications | IAM instance profile | Create (review required)

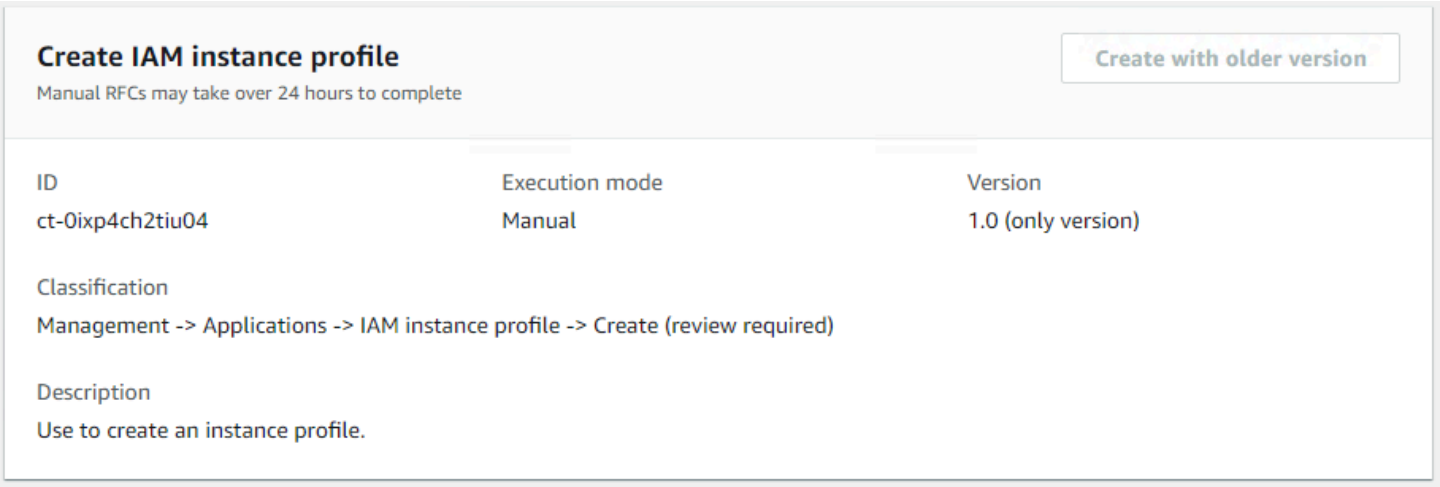
Change Type Details

Change type ID	ct-0ixp4ch2tiu04
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create application IAM instance profile (review required)

Creating IAM instance profiles (review required) with the console



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM instance profiles (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0ixp4ch2tiu04" --change-type-version
 "1.0" --title "TestInstanceProfile" --execution-parameters "{\"InstanceProfileName
\":\"PROFILE_NAME\",\"RelatedIds\":[\"RESOURCE_ID\", \"RESOURCE_ID\"],
\"InstanceProfileDescription\":\"PROFILE_DESCRIPTION\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateInstanceProfileParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0ixp4ch2tiu04"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CreateInstanceProfileParams.json
   ```

2. Modify and save the CreateInstanceProfileParams file. For example, you can replace the contents with something like this:

   ```
   {
     "InstanceProfileDescription":    "PROFILE_DESCRIPTION.",
     "InstanceProfileName":           "PROFILE_NAME",
     "RelatedIds":[\"RESOURCE_ID\", \"RESOURCE_ID\"],
   }
   ```

3. Output the RFC template JSON to a file; this example names it CreateInstanceProfileRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CreateInstanceProfileRfc.json
   ```

4. Modify and save the CreateInstanceProfileRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0ixp4ch2tiu04",
   "Title":                "InstanceProfile-Create-RFC"
   ```

5. Create the RFC, specifying the CreateInstanceProfileRfc file and the CreateInstanceProfileParams file:

```
aws amscm create-rfc --cli-input-json file://CreateInstanceProfileRfc.json  --
execution-parameters file://CreateInstanceProfileParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

For more information about AWS Identity and Access Management, see Using Instance Profiles.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0ixp4ch2tiu04.

Example: Required Parameters

```
{
  "InstanceProfileDescription": "An ample description",
  "InstanceProfileName": "a_good_name"
}
```

Example: All Parameters

```
{
  "InstanceProfileDescription": "An ample description",
  "InstanceProfileName": "a_good_name",
  "RelatedIds": ["foo", "bar", "baz"],
  "Priority": "Medium"
}
```

AWS Backup Subcategory

Change Type Items and Operations in the AWS Backup Subcategory

Backup Job | Start

Start an AWS Backup service backup job to create a one-time snapshot of the specified resource.

Full classification: Management | AWS Backup | Backup job | Start

Change Type Details

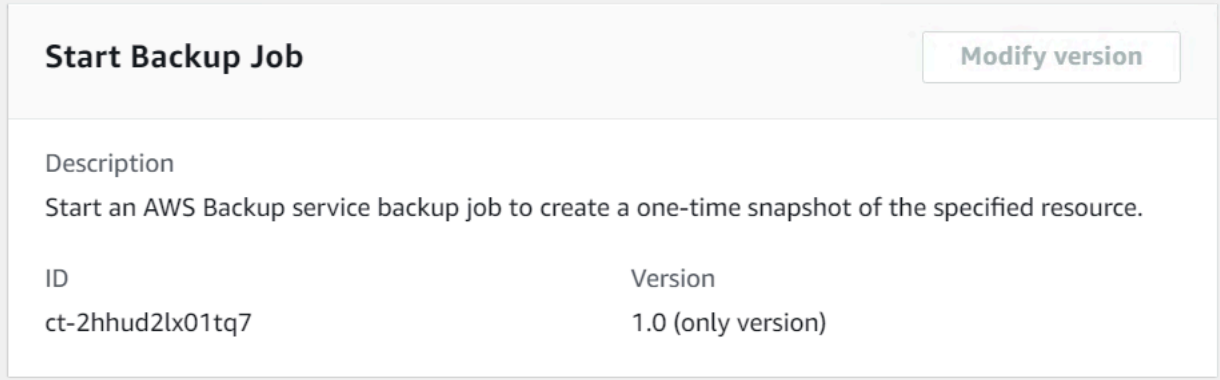
Change type ID	ct-2hhud2lx01tq7
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start AWS Backup job

Starting a backup job with the console

The following shows this change type in the AMS console.



1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting a backup job with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2hhud2lx01tq7" --change-type-version
 "1.0" --title "AWS Backup Start Backup Job" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StartBackupJob\",\"Region\":\"us-east-1\",\"Parameters
\":{\"BackupVaultName\":[\"backup-vault\"],\"ResourceArn\":[\"arn:aws:ec2:us-
east-1:000000000000:volume/vol-123456789\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it StartBackupJobParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-2hhud2lx01tq7" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > StartBackupJobParams.json
```

2. Modify and save the StartBackupJobParams file.

```
{
  "DocumentName": "AWSManagedServices-StartBackupJob",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["backup-vault"],
    "CompleteWindowMinutes": [ "200" ],
    "DeleteAfterDays": [ "10" ],
    "ResourceArn": ["arn:aws:ec2:us-east-1:000000000000:volume/vol-123456789"],
    "StartWindowMinutes": [ "60" ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it StartBackupJobRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StartBackupJobRfc.json
```

4. Modify and save the StartBackupJobRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-2hhud2lx01tq7",
    "ChangeTypeVersion": "1.0",
    "Title": "AWS Backup Start Backup Job"
}
```

5. Create the RFC, specifying the StartBackupJobRfc file and the StartBackupJobParams file:

```
aws amscm create-rfc --cli-input-json file://StartBackupJobRfc.json --execution-
parameters file://StartBackupJobParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2hhud2lx01tq7.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StartBackupJob",
  "Region": "us-east-1",
  "Parameters": {
    "ResourceArn": ["arn:aws:ec2:us-east-1:000000000000:volume/vol-123456789"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StartBackupJob",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": ["backup-vault"],
    "CompleteWindowMinutes": [ "200" ],
    "DeleteAfterDays": [ "10" ],
    "ResourceArn": ["arn:aws:ec2:us-east-1:000000000000:volume/vol-123456789"],
    "StartWindowMinutes": [ "60" ]
  }
}
```

Backup Job | Stop

Stop an AWS Backup service running, or scheduled, backup job.

Full classification: Management | AWS Backup | Backup job | Stop

Change Type Details

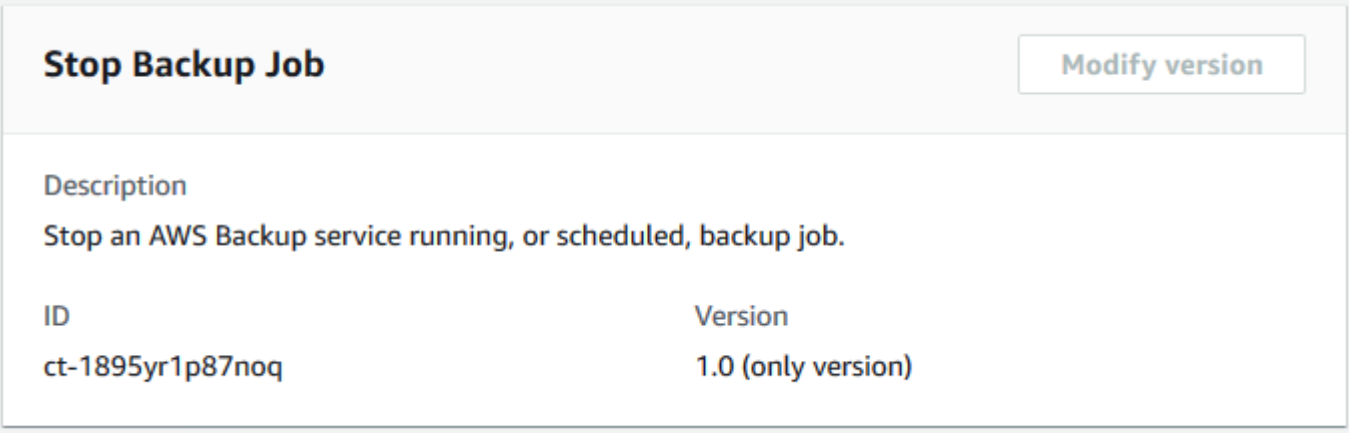
Change type ID	ct-1895yr1p87noq
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop AWS Backup job

Stopping a backup job with the console

The following shows this change type in the AMS console.



1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping a backup job with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1895yr1p87noq" --change-type-version
 "1.0" --title "AWS Backup Stop Backup Job" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-StopBackupJob\",\"Region\":\"us-east-1\",\"Parameters\":
{\"BackupJobId\":[\"278bac28-d634-45b4-85b6-3685e99f2ca1\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it StopBackupJobParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1895yr1p87noq" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopBackupJobParams.json
   ```

2. Modify and save the StopBackupJobParams file.

   ```
   {
       "DocumentName": "AWSManagedServices-StopBackupJob",
       "Region": "us-east-1",
       "Parameters": {
           "BackupJobId": [
               "278bac28-d634-45b4-85b6-3685e99f2ca1"
           ]
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it StopBackupJobRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StopBackupJobRfc.json
```

4. Modify and save the StopBackupJobRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-1895yr1p87noq",
    "ChangeTypeVersion": "1.0",
    "Title": "AWS Backup Stop Backup Job"
}
```

5. Create the RFC, specifying the StopBackupJobRfc file and the StopBackupJobParams file:

```
aws amscm create-rfc --cli-input-json file://StopBackupJobRfc.json --execution-
parameters file://StopBackupJobParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1895yr1p87noq.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-StopBackupJob",
  "Region": "us-east-1",
  "Parameters": {
    "BackupJobId": [ "76659DD5-1A99-46FE-97AD-D6D0126382CA" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-StopBackupJob",
  "Region": "us-east-1",
  "Parameters": {
    "BackupJobId": [ "2abf2ce0-9096-407c-95a6-4d0b584b9a0a" ]
  }
}
```

Backup Plan | Enable Cross Account Copy (Management Account)

Enable and configure cross-account backup and monitoring in a management account. This automation can only be completed successfully in a management account.

Full classification: Management | AWS Backup | Backup plan | Enable cross account copy (Management account)

Change Type Details

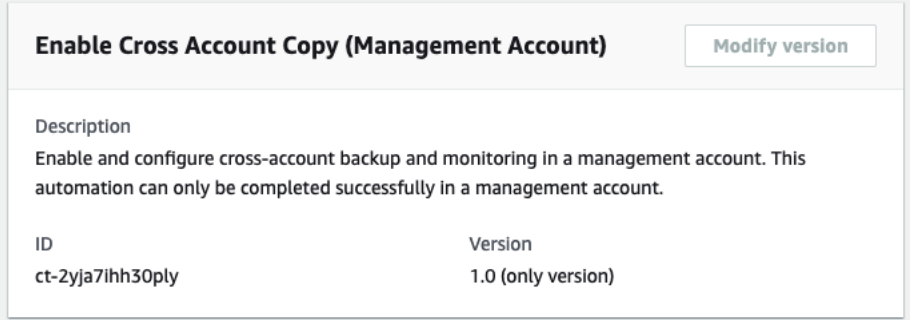
Change type ID	ct-2yja7ihh30ply
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable cross account backup plan copy

Enabling cross account backup plan copy with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling cross account backup plan copy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-2yja7ihh30ply" \
--change-type-version "1.0" --title "ConfigCrossAccountCopyInManagementAccount" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
HandleConfigureCrossAccountBackupInManagementAccount-Admin\",\"Region\":\"ap-
```

```
southeast-2\",\"Parameters\":{\"DestinationAccountId\":[\"123456789012\"],
\"SourceAccountId\":[\"210987654321\"]}}" \
--endpoint-url https://amscm-gamma.us-east-1.amazonaws.com
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it ConfigCrossAcctCopyBackupPlanParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-2yja7ihh30ply"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 ConfigCrossAcctCopyBackupPlanParams.json
```

2. Modify and save the ConfigCrossAcctCopyBackupPlanParams file.

```
{
  "DocumentName": "AWSManagedServices-
HandleConfigureCrossAccountBackupInManagementAccount-Admin",
  "Region": "ap-southeast-2",
  "Parameters": {
      "DestinationAccountId": [
        "123456789012"
      ],
     "SourceAccountId": [
       "210987654321"
      ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ConfigCrossAcctCopyBackupPlanRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton >
 ConfigCrossAcctCopyBackupPlanRfc.json
```

4. Modify and save the ConfigCrossAcctCopyBackupPlanRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-2yja7ihh30ply",
  "ChangeTypeVersion": "1.0",
  "Title": "ConfigureCrossAcctCopyBackup"
```

```
}
```

5. Create the RFC, specifying the ConfigCrossAcctCopyBackupPlanRfc file and the ConfigCrossAcctCopyBackupPlanParams file:

```
aws amscm create-rfc --cli-input-json file://ConfigCrossAcctCopyBackupPlanRfc.json
 --execution-parameters file://ConfigCrossAcctCopyBackupPlanParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2yja7ihh30ply.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-
HandleConfigureCrossAccountBackupInManagementAccount-Admin",
    "Region": "us-east-1",
    "Parameters": {
      "DestinationAccountId": [ "123456789012" ],
      "SourceAccountId": [ "123456789012" ]
    }
}
```

Backup Plan | Enable Cross Region Copy

Update an existing backup plan rule with copy actions like cross region destination vault, and storage retention settings.

Full classification: Management | AWS Backup | Backup plan | Enable cross region copy

Change Type Details

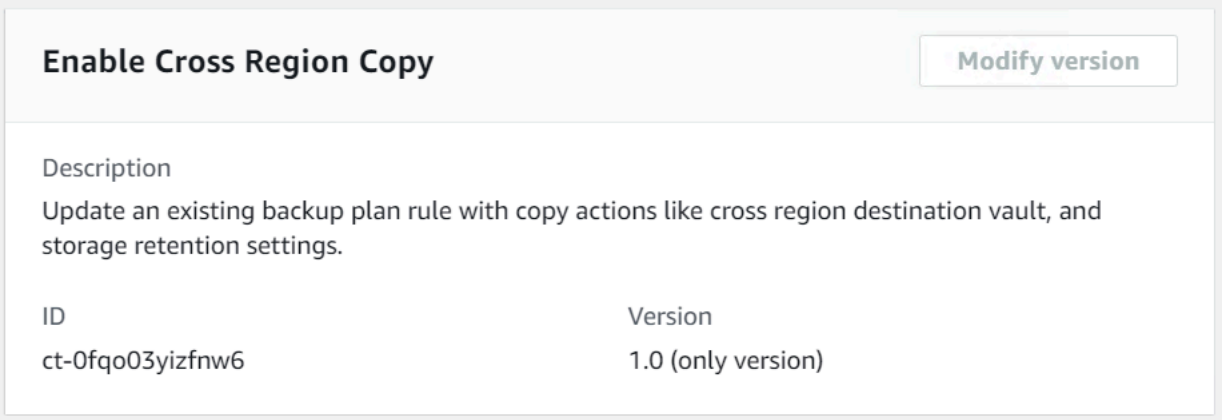
Change type ID	ct-0fqo03yizfnw6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enabling a cross region backup plan copy

Enabling a cross-region backup plan copy with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling a cross-region backup plan copy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-0fqo03yizfnw6" \
--change-type-version "1.0" --title "ConfigureCrossRegionBackup" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ConfigureCrossRegionBackup\",\"Region\":\"us-east-1\",\"Parameters\":{\"BackupPlanName
\":[\"ConfigureCrossRegionBackup-Plan\"],\"RuleName\":[\"BackupRule1\"],
\"DestinationRegion\":[\"eu-west-1\"],\"DestinationVaultName\":[\"vault-
test-ConfigureCrossRegionBackup\"],\"DeleteAfterNumberOfDays\":[\"250\"],
\"MoveToColdStorageAfterNumberOfDays\":[\"150\"]}}" \
--endpoint-url https://amscm-gamma.us-east-1.amazonaws.com
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it ConfigCrossRegionBackupPlanParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0fqo03yizfnw6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ConfigCrossRegionBackupPlanParams.json
   ```

2. Modify and save the ConfigCrossRegionBackupPlanParams file.

```
{
  "DocumentName": "AWSManagedServices-ConfigureCrossRegionBackup",
  "Region": "us-east-1",
  "Parameters": {
      "BackupPlanName": [ "ConfigureCrossRegionBackup-Plan"],
      "RuleName": ["BackupRule1"],
      "DestinationRegion": [ "eu-west-1"],
      "DestinationVaultName": [ "vault-test-ConfigureCrossRegionBackup"],
      "DeleteAfterNumberOfDays": [ "250"],
      "MoveToColdStorageAfterNumberOfDays": [ "150"]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it ConfigCrossRegionBackupPlanRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > ConfigCrossRegionBackupPlanRfc.json
```

4. Modify and save the ConfigCrossRegionBackupPlanRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-0fqo03yizfnw6",
  "ChangeTypeVersion": "1.0",
  "Title": "ConfigureCrossRegionBackup"
}
```

5. Create the RFC, specifying the ConfigCrossRegionBackupPlanRfc file and the ConfigCrossRegionBackupPlanParams file:

```
aws amscm create-rfc --cli-input-json file://ConfigCrossRegionBackupPlanRfc.json --
execution-parameters file://ConfigCrossRegionBackupPlanParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0fqo03yizfnw6.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-ConfigureCrossRegionBackup",
  "Region": "us-east-1",
  "Parameters": {
    "BackupPlanName": [ "backup-vault" ],
    "RuleName": [ "Daily-Backup" ],
    "DestinationRegion": [ "eu-west-1" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ConfigureCrossRegionBackup",
  "Region": "us-east-1",
  "Parameters": {
    "BackupPlanName": [ "backup-vault" ],
    "RuleName": [ "Daily-Backup" ],
    "DestinationRegion": [ "eu-west-1" ],
    "DestinationVaultName": [ "ams-replication-vault" ],
    "DestinationEncryptionKeyArn": ["arn:aws:kms:us-
east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"],
    "DeleteAfterNumberOfDays": [ "100" ],
    "MoveToColdStorageAfterNumberOfDays": [ "200" ]
  }
}
```

Backup Plan | Update (Review Required)

Update an existing backup plan. Please note that any changes that you make to a backup plan have no effect on existing backups created by the backup plan. The changes apply only to backups that are created in the future.

Full classification: Management | AWS Backup | Backup plan | Update (review required)

Change Type Details

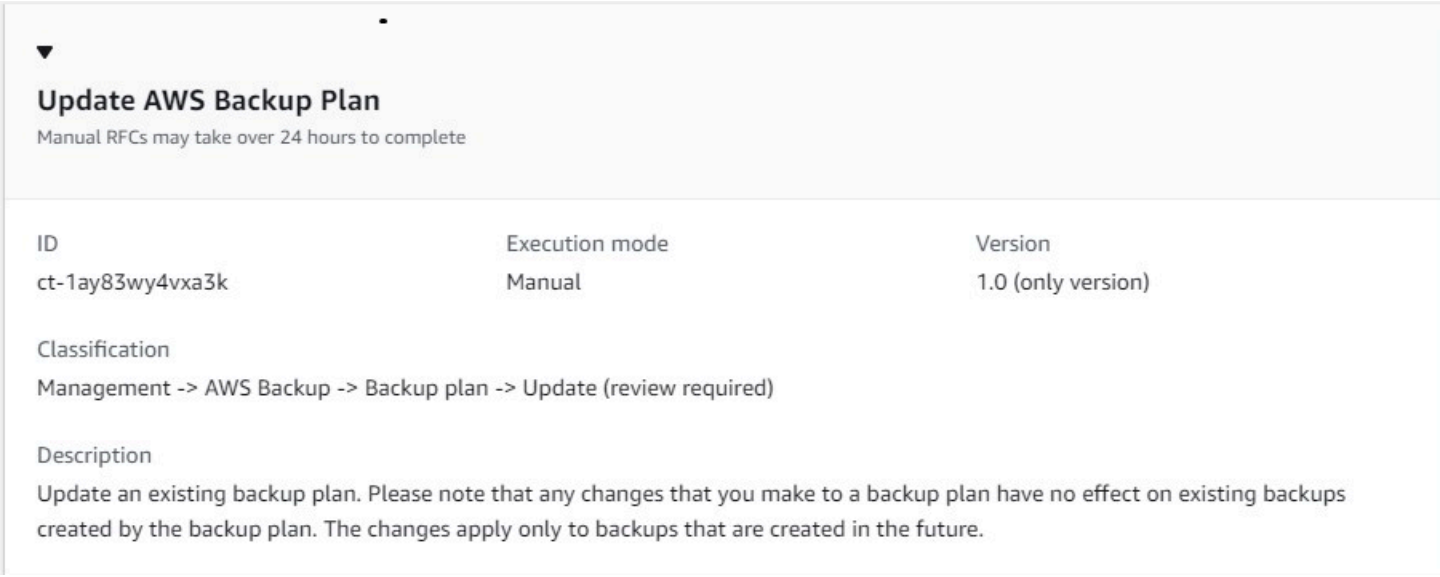
Change type ID	ct-1ay83wy4vxa3k
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update AWS Backup plan (review required)

Updating an AWS Backup plan with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AWS Backup plan with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1ay83wy4vxa3k" --change-type-version
 "1.0" --title "Update AWSBackup Plan" --execution-parameters ""{\"BackupPlanName
\":\"PLAN_NAME\",\"ResourceTagKey\":\"TAG_KEY\",\"ResourceTagValue\":
\"TAG_VALUE\",\"BackupRuleName\":\"RULE_NAME\",\"BackupRuleVault\":\"VAULT\",
\"BackupRuleCompletionWindowMinutes\":120,\"BackupRuleScheduleExpression\":\"cron(0
 1 ? * * *)\",\"BackupRuleDeleteAfterDays\":90,\"BackupRuleMoveToColdStorageAfterDays
\":365,\"BackupRuleStartWindowMinutes\":60,\"BackupRuleRecoveryPointTagKey
\":\"TAG_KEY\",\"BackupRuleRecoveryPointTagValue\":\"TAG_VALUE\,
\"BackupRuleEnableContinuousBackup\":\"false\",\"BackupRuleCopyActionsDestVaultArn
\":\"VAULT\",\"BackupRuleCAMoveToColdStorageAfterDays\":0,
\"BackupRuleCopyActionsDeleteAfterDays\":90}""
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdateBackupPlanParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1ay83wy4vxa3k"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 UpdateBackupPlanParams.json
```

2. Modify and save the UpdateBackupPlanParams file.

```
{
  "BackupPlanName": "MyCustomBackupPlan",
  "ResourceTagKey": "custom_backup_test",
  "ResourceTagValue": "true",
  "WindowsVSS": "disabled",
  "BackupRuleName": "BackupRule",
  "BackupRuleVault": "ams-custom-backups",
  "BackupRuleCompletionWindowMinutes": 1440,
  "BackupRuleScheduleExpression": "cron(0 2 ? * * *)",
  "BackupRuleDeleteAfterDays": 0,
  "BackupRuleMoveToColdStorageAfterDays": 0,
  "BackupRuleStartWindowMinutes": 180,
  "BackupRuleRecoveryPointTagKey": "test",
  "BackupRuleRecoveryPointTagValue": "test",
  "BackupRuleEnableContinuousBackup": "false",
  "BackupRuleCopyActionsDestVaultArn": "",
  "BackupRuleCAMoveToColdStorageAfterDays": 0,
  "BackupRuleCopyActionsDeleteAfterDays": 0
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdateBackupPlanRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateBackupPlanRfc.json
```

4. Modify and save the UpdateBackupPlanRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1ay83wy4vxa3k",
  "Title": "Update AWS Backup Plan"
}
```

5. Create the RFC, specifying the UpdateBackupPlanRfc file and the UpdateBackupPlanParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateBackupPlanRfc.json --execution-
parameters file://UpdateBackupPlanParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> Not all resource types supported by AWS Backup are enabled by default. Review the enabled resource types in your account using Getting Started 1: Service Opt-In.

To learn more about AWS Backup, see AWS Backup: How It Works.

Before creating backup plans, confirm supported resources at Feature availability by resource.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1ay83wy4vxa3k.

Example: Required Parameters

```
{
  "BackupPlanName": "MyCustomBackupPlan",
  "BackupRuleName": "BackupRule",
  "BackupRuleVault": "ams-custom-backups"
}
```

Example: All Parameters

```
{
  "BackupPlanName": "MyCustomBackupPlan",
  "ResourceTagKey": "custom_backup_test",
```

```
  "ResourceTagValue": "true",
  "WindowsVSS": "disabled",
  "BackupRuleName": "BackupRule",
  "BackupRuleVault": "ams-custom-backups",
  "BackupRuleCompletionWindowMinutes": 1440,
  "BackupRuleScheduleExpression": "cron(0 2 ? * * *)",
  "BackupRuleDeleteAfterDays": 0,
  "BackupRuleMoveToColdStorageAfterDays": 0,
  "BackupRuleStartWindowMinutes": 180,
  "BackupRuleRecoveryPointTagKey": "test",
  "BackupRuleRecoveryPointTagValue": "test",
  "BackupRuleEnableContinuousBackup": "false",
  "BackupRuleCopyActionsDestVaultArn": "",
  "BackupRuleCAMoveToColdStorageAfterDays": 0,
  "BackupRuleCopyActionsDeleteAfterDays": 0,
  "Priority": "Medium"
}
```

Recovery Point | Delete

Delete one or more recovery points (snapshots) from the specified vault. Use this change type to delete recovery points that were manually created, and recovery points that were created through a backup plan, and that are older than 30 days. The deletion of recovery points cannot be rolled back.

Full classification: Management | AWS Backup | Recovery point | Delete

Change Type Details

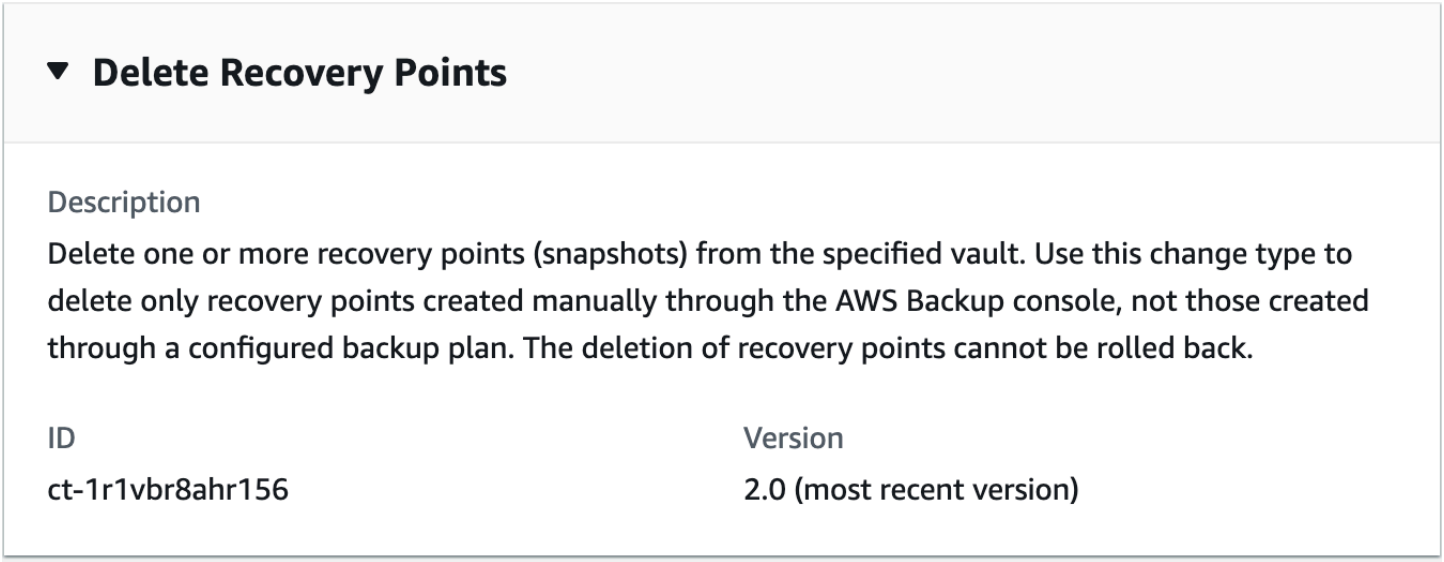
Change type ID	ct-1r1vbr8ahr156
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete AWS Backup recovery points

Deleting recovery points with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting recovery points with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1r1vbr8ahr156" --change-type-version "2.0"
 --title "AWS Backup Delete Recovery Points" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-DeleteRecoveryPoints\",\"Region\":\"us-east-1\",\"Parameters\":
{\"BackupVaultName\":[\"ams-manual-backups\"],\"RecoveryPointArns\":[\"arn:aws:ec2:us-
east-1::snapshot/snap-000000000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it DeleteRecoveryPointsParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1r1vbr8ahr156"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteRecoveryPointsParams.json
   ```

2. Modify and save the DeleteRecoveryPointsParams file.

   ```
   {
       "DocumentName": "AWSManagedServices-DeleteRecoveryPoints",
       "Region": "us-east-1",
       "Parameters": {
           "BackupVaultName": [
               "ams-manual-backups"
            ],
           "RecoveryPointArns": [
               "arn:aws:backup:us-east-1:000000000000:recovery-
   point:24f48ec5-79a7-4a40-b992-d97583518f2f",
               "arn:aws:backup:us-east-1:000000000000:recovery-point:3b6a599e-
   b5a3-4028-87b3-be9a1fdc01e8"
           ]
       }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it DeleteRecoveryPointsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteRecoveryPointsRfc.json
```

4. Modify and save the DeleteRecoveryPointsRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-1r1vbr8ahr156",
    "ChangeTypeVersion": "2.0",
    "Title": "AWS Backup Delete Recovery Points"
}
```

5. Create the RFC, specifying the DeleteRecoveryPointsRfc file and the DeleteRecoveryPointsParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteRecoveryPointsRfc.json --
execution-parameters file://DeleteRecoveryPointsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This CT is now at version 2.0. This reflects development to allow you to delete more than one recovery point at a time.

To learn more about AWS Backup, see AWS Backup: How It Works.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1r1vbr8ahr156.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteRecoveryPoints",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": [ "backup-vault" ],
    "RecoveryPointArns": [ "arn:aws:backup:us-east-1:123456789012:recovery-
point:1EB3B5E7-9EB0-435A-A80B-108B488B0D45", "arn:aws:backup:us-
east-1:123456789012:recovery-point:1EB3B5E7-9EB0-435A-A80B-108B488B0D41" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeleteRecoveryPoints",
  "Region": "us-east-1",
  "Parameters": {
    "BackupVaultName": [ "backup-vault" ],
    "RecoveryPointArns": [ "arn:aws:backup:us-east-1:123456789012:recovery-
point:1EB3B5E7-9EB0-435A-A80B-108B488B0D45", "arn:aws:backup:us-
east-1:123456789012:recovery-point:1EB3B5E7-9EB0-435A-A80B-108B488B0D41" ]
  }
}
```

AWS Service Subcategory

Change Type Items and Operations in the AWS Service Subcategory

- Self-Provisioned Service | Add
- Self-Provisioned Service | Add (Review Required)

Self-Provisioned Service | Add

Add a specific, allowed, AWS service to your AMS account. This CT validates prerequisites in the account and deploys a service with the default parameters. Not all Self-service provisioning services are supported, the ServiceName parameter for this CT lists the ones that are. For each service that you add, AMS creates a new role so you use the service without AMS management under the AMS Shared Responsibility model. Compliance is a shared responsibility and your

AMS compliance status does not automatically apply to services or applications that you add in this way. Some AWS services do not have compliance certifications. For more information, see the AWS Services in Scope of AWS Assurance Program page. On that page, unless specifically excluded, features of each of the services are considered in scope of the assurance programs, and are reviewed and tested as part of our assessment when you submit this CT.

Full classification: Management | AWS service | Self-provisioned service | Add

Change Type Details

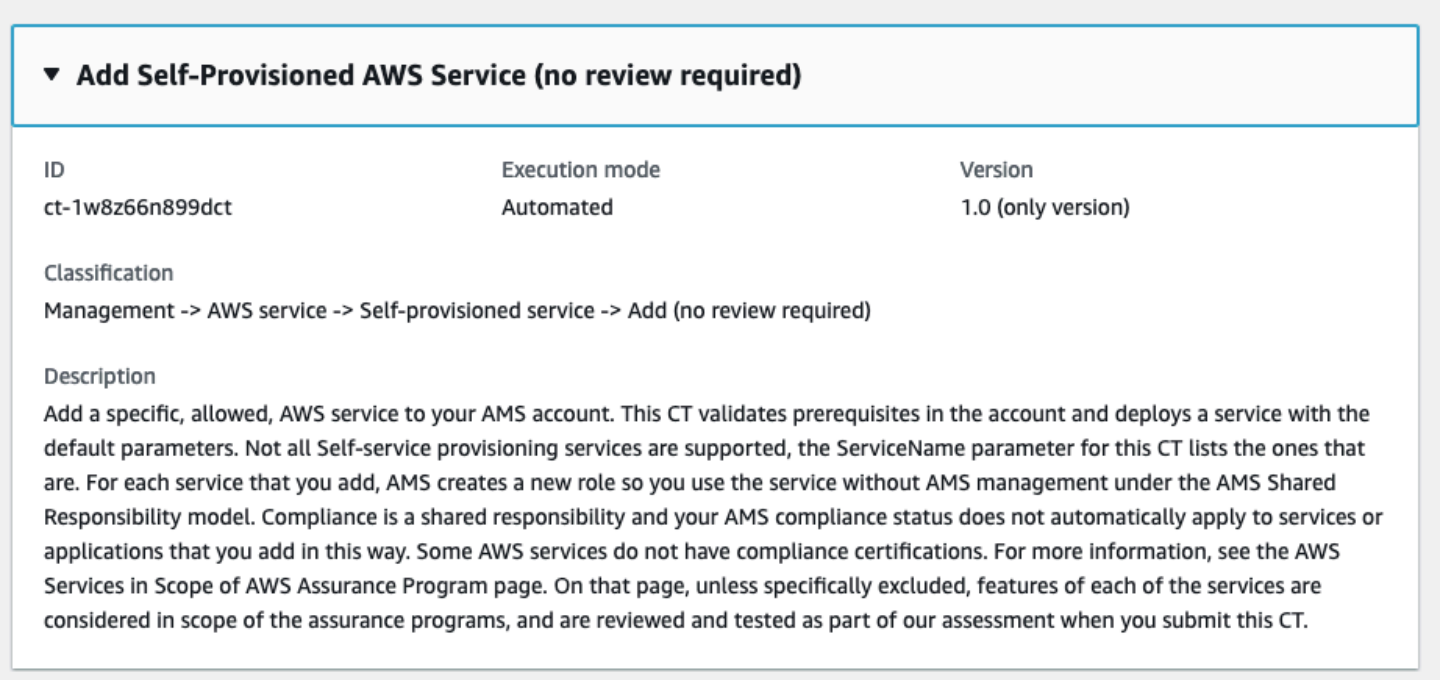
Change type ID	ct-1w8z66n899dct
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add Self-Service Provisioning service

Adding an AMS self-provisioned AWS service with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AMS self-provisioned AWS service with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1w8z66n899dct" --change-type-version "1.0"
 --title "Add new Self-provisioned service" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-HandleCreateSSPSResources-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"ServiceName\": \"AWS License Manager\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named AddSspsParameters.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1w8z66n899dct" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddSspsParameters.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-HandleCreateSSPSResources-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "ServiceName": "AWS License Manager"
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it AddSspsRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AddSspsRfc.json
   ```

4. Modify and save the AddSspsRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":          "ct-1w8z66n899dct",
   "ChangeTypeVersion":     "1.0",
   ```

```
"Title":                "Add new Self-provisioned service"
}
```

5. Create the RFC, specifying the SelfServeServiceRfc file and the SelfServeServiceParams file:

```
aws amscm create-rfc --cli-input-json file://AddSspsRfc.json  --execution-
parameters file://AddSspsParameters.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- The **ServiceName** field in the console lists the self-provisioned services supported by this CT. Limited to AMS-approved AWS services. For a list, see Setting Up Self-serve Services.
- For self-provisioned services not supported by this CT, or for deployment with custom parameters, use the manual version: Management | AWS service | Self-Provisioned Service | Add (review required) (ct-3qe6io8t6jtny).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1w8z66n899dct.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateSSPSResources-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "ServiceName": "AWS License Manager"
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-HandleCreateSSPSResources-Admin",
  "Region" : "us-east-1",
  "Parameters": {
```

```
    "ServiceName": "AWS License Manager",
    "SAMLProviders": "foo-saml-provider,bar-saml-provider",
    "IAMRole": "testing_role"
  }
}
```

Self-Provisioned Service | Add (Review Required)

Add a specific, allowed, AWS service to your AMS account. AMS adds the necessary permissions to use the service to an existing IAM role that you specify, or creates a new role that allows you to use the service without AMS management under the AMS Shared Responsibility model. Compliance is a shared responsibility and your AMS compliance status does not automatically apply to services or applications that you add in this way. Some AWS services do not have compliance certifications. For more information, go to the AWS Services in Scope of AWS Assurance Program page. On that page, unless specifically excluded, features of each of the services are considered in scope of the assurance programs, and are reviewed and tested as part of the assessment.

Full classification: Management | AWS service | Self-provisioned service | Add (review required)

Change Type Details

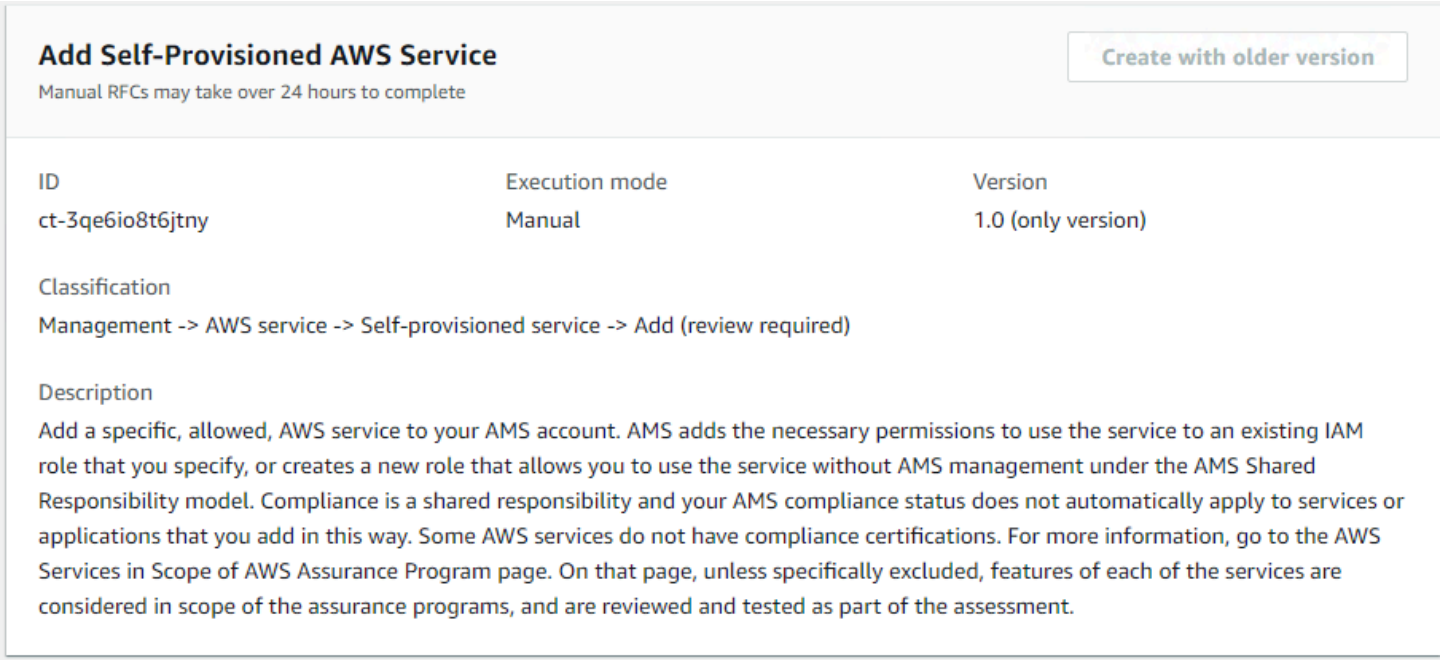
Change type ID	ct-3qe6io8t6jtny
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Add Self-Service Provisioning service (review required)

Adding an AMS self-provisioned AWS service with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AMS self-provisioned AWS service with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

All parameters:

```
aws amscm create-rfc --title Add-Self-Serve-Service --change-type-id ct-3qe6io8t6jtny
 --change-type-version 1.0 --execution-parameters '{"ServiceName":"AWS Certificate
 Manager (ACM)","IAMRole":"arn:aws:iam::123456789012:role/customer_security_role",
 "SAMLProviders":"SAML_PROVIDER,SAML_PROVIDER"}'
```

Only required parameters:

```
aws amscm create-rfc --title add-self-serve-service --change-type-id ct-3qe6io8t6jtny
 --change-type-version 1.0 --execution-parameters '{"ServiceName":"AWS License
 Manager"}'
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named SelfServeServiceParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3qe6io8t6jtny"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    SelfServeServiceParams.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
   "ServiceName":   "AWS Certificate Manager (ACM)",
   "IAMRole":       "arn:aws:iam::123456789012:role/customer_security_role",
   "SAMLProviders": "SAML_PROVIDER, SAML_PROVIDER"
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it SelfServeServiceRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > SelfServeServiceRfc.json
   ```

4. Modify and save the SelfServeServiceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-3qe6io8t6jtny",
"ChangeTypeVersion":     "1.0",
"Title":                 "Self-Serve-Service-RFC"
}
```

5. Create the RFC, specifying the SelfServeServiceRfc file and the SelfServeServiceParams file:

```
aws amscm create-rfc --cli-input-json file://SelfServeServiceRfc.json  --execution-
parameters file://SelfServeServiceParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For automated deployment of most self-provisioned services with default roles, use: Management | AWS service | Self-provisioned service | Add (no review required) (ct-1w8z66n899dct). See Add Self-Service Provisioning service. UUse this "review required" change type (ct-3qe6io8t6jtny) for services not supported by ct-1w8z66n899dct or for deployments with custom parameters.
- For a list of which self-provisioned services you can add using CloudFormation Ingest, see CloudFormation Ingest stack: supported resources.
- This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.
- The **ServiceName** parameter is limited to AMS-approved AWS services. For a list, see Setting Up Self-serve Services.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3qe6io8t6jtny.

Example: Required Parameters

```
{
  "ServiceName": "AWS License Manager"
}
```

Example: All Parameters

```
{
  "ServiceName": "AWS License Manager",
  "IAMRole": "arn:aws:iam::123456789012:role/myrole",
  "SAMLProviders": "foo-saml-provider",
  "Priority": "Medium"
}
```

Custom Stack Subcategory

Change Type Items and Operations in the Custom Stack Subcategory

- Stack from CloudFormation Template | Approve Changeset and Update
- Stack from CloudFormation Template | Remediate Drift
- Stack from CloudFormation Template | Remediate Drift (Review Required)
- Stack from CloudFormation Template | Update

Stack from CloudFormation Template | Approve Changeset and Update

Approve and execute an existing ChangeSet to update a CloudFormation stack. This ChangeType is used primarily to approve and apply changes requested using the "Update CloudFormation stack" CT that would cause removal or replacement of resources, but can also be used to execute any existing ChangeSet to update CloudFormation stacks.

Full classification: Management | Custom Stack | Stack from CloudFormation Template | Approve Changeset and Update

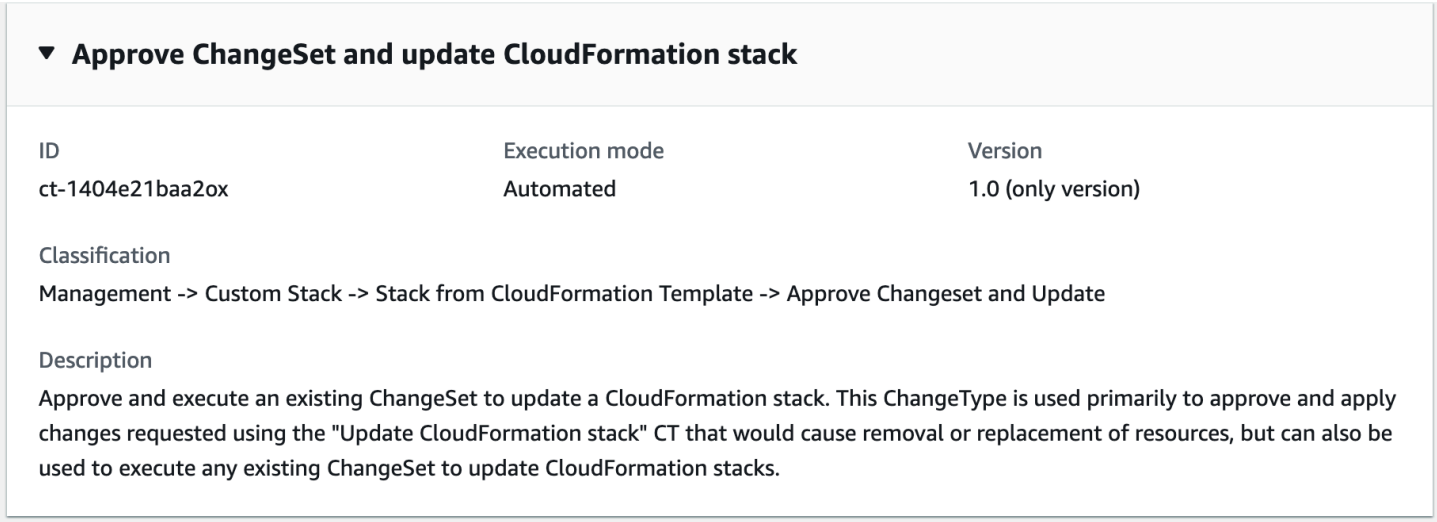
Change Type Details

Change type ID	ct-1404e21baa2ox
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Approve a CloudFormation ingest stack changeset

Approving and updating a CloudFormation ingest stack using the console



To approve and update a CloudFormation ingest stack using the console

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Approving and updating a CloudFormation ingest stack using the CLI

To approve and update a CloudFormation ingest stack using the CLI

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

1. Output the execution parameters JSON schema for this change type to a file in your current folder. This example names it CreateAsgParams.json:

```
aws amscm create-rfc --change-type-id "ct-1404e21baa2ox" --change-
type-version "1.0" --title "Approve Update" --execution-parameters
 file://PATH_TO_EXECUTION_PARAMETERS --profile saml
```

2. Modify and save the schema as follows:

```
{
  "StackId": "STACK_ID",
  "VpcId": "VPC_ID",
  "ChangeSetName": "UPDATE-ef81e2bc-03f6-4b17-a3c7-feb700e78faa",
  "TimeoutInMinutes": 1080
}
```

Tips

Note

If there are multiple resources in a stack, and you want to delete only a subset of the stack resources, use the CloudFormation Update CT; see CloudFormation Ingest Stack: Updating. You can also submit a Management | Other | Other | Update change type and AMS engineers can help you craft the changeset, if needed.

To learn more about AWS CloudFormation, see AWS CloudFormation.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1404e21baa2ox.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Stack from CloudFormation Template | Remediate Drift

Remediate the drift (out-of-band changes) in a stack, bringing the stack in sync and enabling you to perform future updates using the available Update CTs. Note: up to 10 drifted resources will be remediated per RFC.

Full classification: Management | Custom Stack | Stack from CloudFormation Template | Remediate drift

Change Type Details

Change type ID	ct-3kinq0u4l33zf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remediate stack drift

Remediating Stack Drift with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Remediating Stack Drift with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3kinq0u4l33zf" --change-type-version "1.0" --
title "Remediate Stack Drift, no ops review" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StartDriftRemediation\", \"Region\": \"us-east-1\", \"Parameters
\": {\"StackName\": [\"stack-xxxxxxxxxxxxxxxxx\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemediateDriftNrrParams.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftNrrParams.json
   ```

2. Modify and save the RemediateDriftNrrParams file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-StartDriftRemediation",
       "Region": "us-east-1",
       "Parameters": {
           "StackName": [
               "stack-xxxxxxxxxxxxxxxxx"
           ]
       }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it RemediateDriftNrrRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftNrrRfc.json
   ```

4. Modify and save the RemediateDriftNrrRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId": "ct-3kinq0u4l33zf",
   "ChangeTypeVersion": "1.0",
   ```

```
"Title": "Remediate stack drift, no ops review"
}
```

5. Create the RFC, specifying the RemediateDriftNrrRfc file and the RemediateDriftNrrParams file:

```
aws amscm create-rfc --cli-input-json file://RemediateDriftNrrRfc.json  --
execution-parameters file://RemediateDriftNrrParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Important

Stack remediation modifies the stack template and/or parameter values. Once remediation is complete, you must update your local template repositories, or any automation, that would be updating the remediated stack, with the latest template and parameters provided in the RFC summary of the remeditation. It is very important to do this, because using the old template and/or parameters can cause destructive changes on the stack resources. For more details, including a list of *Limitations*, see Drift remediation FAQs.

Note

When using "review required" CTs, AMS recommends that you use the ASAP **Scheduling** option (choose **ASAP** in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run. If you schedule these RFCs, be sure to allow at least 24 hours. If approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3kinq0u4l33zf.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StartDriftRemediation",
    "Region": "us-east-1",
    "Parameters": {
      "StackName": ["stack-a1b2c3d4e5f678900"],
      "DryRun": ["true"]
    }
}
```

Stack from CloudFormation Template | Remediate Drift (Review Required)

Remediate the drift (out-of-band changes) in a stack, bringing the stack in sync and enabling you to perform future updates using the available Update CTs. Drift remediation can be performed on EC2 resource types.

Full classification: Management | Custom Stack | Stack from CloudFormation Template | Remediate drift (review required)

Change Type Details

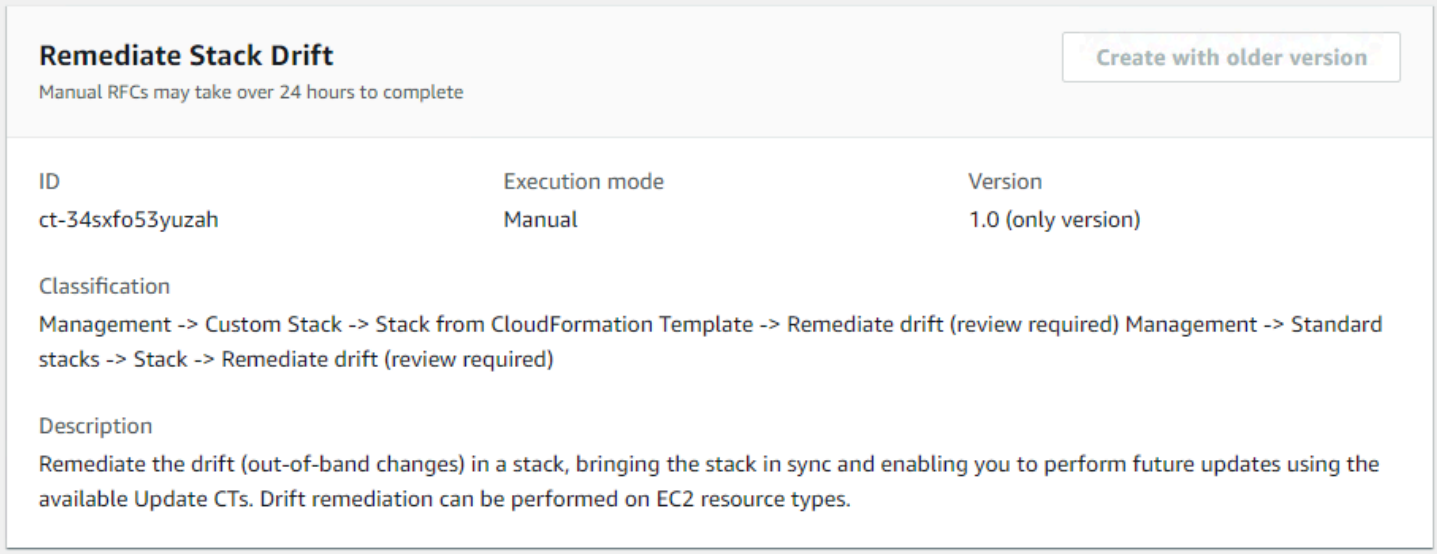
Change type ID	ct-34sxfo53yuzah
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Remediate stack drift (review required)

Remediating Stack Drift (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Remediating Stack Drift (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-34sxfo53yuzah" --change-type-version
 "1.0" --title "Remediate stack drift" --execution-parameters '{"StackName":"stack-
a1b2c3d4e5f67890e","DryRun":false}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemediateDriftParams.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftParams.json
   ```

2. Modify and save the RemediateDriftParams file. For example, you can replace the contents with something like this:

   ```
   {
   "StackName" : "stack-a1b2c3d4e5f67890e",
   "DryRun" : false
   }
   ```

3. Output the RFC template JSON file to a file; this example names it RemediateDriftRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftRfc.json
   ```

4. Modify and save the RemediateDriftRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId": "ct-34sxfo53yuzah",
   "ChangeTypeVersion": "1.0",
   "Title": "Remediate stack drift"
   ```

```
}
```

5. Create the RFC, specifying the RemediateDriftRfc file and the RemediateDriftParams file:

```
aws amscm create-rfc --cli-input-json file://RemediateDriftRfc.json  --execution-
parameters file://RemediateDriftParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

> **Note**
>
> When using "review required" CTs, AMS recommends that you use the ASAP **Scheduling** option (choose **ASAP** in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run. If you schedule these RFCs, be sure to allow at least 24 hours. If approval does not happen before the scheduled start time, the RFC is rejected automatically.

- There is an automated version of this change type that runs more quickly, though there are some limitations. For more details, see Stack | Remediate Drift.
- Stack remediation modifies the stack template and/or parameter values. Once remediation is complete, you must update your local template repositories, or any automation, that would be updating the remediated stack, with the latest template and parameters provided in the RFC summary of the remeditation. It is very important to do this, because using the old template and/or parameters can cause destructive changes on the stack resources.

  For more details, see Drift remediation FAQs.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-34sxfo53yuzah.

Example: Required Parameters

```
{
  "StackName": "stack-a1b2c3d4e5f678900"
}
```

Example: All Parameters

```
{
  "StackName": "stack-a1b2c3d4e5f678900",
  "DryRun": false,
  "Priority": "Medium"
}
```

Stack from CloudFormation Template | Update

Update the template and/or parameters of a CFN stack. To only update the parameters in an existing stack a modified CFN template is not required, modified parameters can be provided instead. Values for existing parameters are overwritten, values for new parameters are added. To add, delete or modify a resource, or to change attributes not referenced through a parameter, use a modified CFN template. If the update would result in a resource in the stack being replaced or removed, the RFC fails and requires approval through the "Approve ChangeSet and update CloudFormation stack" CT (ct-1404e21baa2ox).

Full classification: Management | Custom Stack | Stack from CloudFormation Template | Update

Change Type Details

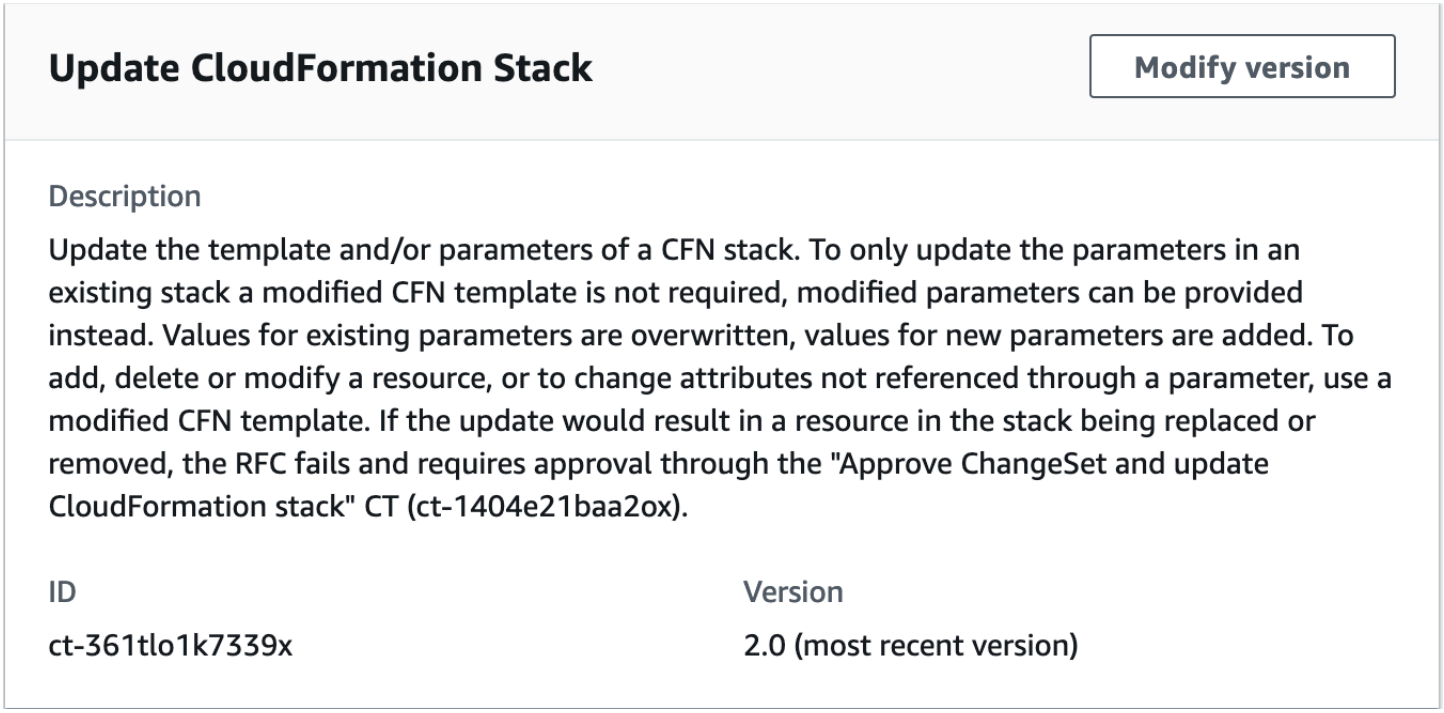
Change type ID	ct-361tlo1k7339x
Current version	2.0
Expected execution duration	360 minutes
AWS approval	Required

Customer approval	Not required
Execution mode	Automated

Additional Information

Update AWS CloudFormation ingest stack

Updating a CloudFormation ingest stack using the console



To update a CloudFormation Ingest Stack using the console

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a CloudFormation ingest stack using the CLI

To update a CloudFormation ingest stack using the CLI

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

1. Prepare the AWS CloudFormation template that you want to use to update the stack, and upload it to your S3 bucket. For important details, see AWS CloudFormation Ingest Guidelines, Best Practices, and Limitations.
2. Create and submit the RFC to AMS:
   - Create and save the execution parameters JSON file, include the CloudFormation template parameters that you want. This example names it UpdateCfnParams.json.

     Example UpdateCfnParams.json file with inline parameter updates:

```
{
  "StackId": "stack-yjjoo9aicjyqw4ro2",
  "VpcId": "VPC_ID",
  "CloudFormationTemplate": "{\"AWSTemplateFormatVersion\":\"2010-09-09\",
\"Description\":\"Create a SNS topic\",\"Parameters\":{\"TopicName\":{\"Type
\":\"String\"},\"DisplayName\":{\"Type\":\"String\"}},\"Resources\":{\"SnsTopic
\":{\"Type\":\"AWS::SNS::Topic\",\"Properties\":{\"TopicName\":{\"Ref\":
\"TopicName\"},\"DisplayName\":{\"Ref\":\"DisplayName\"}}}}}",
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1440
}
```

Example UpdateCfnParams.json file with S3 bucket endpoint containing an updated CloudFormation template:

```
{
  "StackId": "stack-yjjoo9aicjyqw4ro2",
  "VpcId": "VPC_ID",
  "CloudFormationTemplateS3Endpoint": "s3_url",
  "TemplateParameters": [
    {
      "Key": "TopicName",
      "Value": "TopicNameCLI"
    },
    {
      "Key": "DisplayName",
      "Value": "DisplayNameCLI"
    }
  ],
  "TimeoutInMinutes": 1080
}
```

3. Create and save the RFC parameters JSON file with the following content. This example names it UpdateCfnRfc.json file.

```
{
   "ChangeTypeId": "ct-361tlo1k7339x",
   "ChangeTypeVersion": "1.0",
   "Title": "cfn-ingest-template-update"
}
```

4. Create the RFC, specifying the UpdateCfnRfc file and the UpdateCfnParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateCfnRfc.json  --execution-
parameters file://UpdateCfnParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- This change type is now at version 2.0. Changes include removing the **AutoApproveUpdateForResources** parameter, which was used in version 1.0 of this CT, and adding two new parameters: **AutoApproveRiskyUpdates** and **BypassDriftCheck**.
- If the S3 bucket exists in an AMS account, you must use your AMS credentials for this command. For example, you may need to append `--profile saml` after obtaining your AMS AWS Security Token Service (AWS STS) credentials.
- All `Parameter` values for resources in the CloudFormation template must have a value, either through a default or a custom value through the parameters section of the CT. You can override the parameter value by structuring the CloudFormation template resources to reference a Parameters key. For examples that show how to do, see CloudFormation ingest stack: CFN validator examples.

  IMPORTANT: Missing parameters not supplied explicitly in the form, default to the currently set values on the existing stack or template.
- For a list of which self-provisioned services you can add using AWS CloudFormation Ingest, see CloudFormation Ingest Stack: Supported Resources.

  To learn more about AWS CloudFormation, see AWS CloudFormation.

Validating a AWS CloudFormation ingest

The template is validated to ensure that it can be created in an AMS account. If it passes validation, it's updated to include any resources or configurations required for it to conform with AMS. This includes adding resources such as Amazon CloudWatch alarms in order to allow AMS Operations to monitor the stack.

The RFC is rejected if any of the following are true:

- RFC JSON Syntax is incorrect or does not follow the given format.
- The provided S3 bucket presigned URL is not valid.
- The template is not valid AWS CloudFormation syntax.
- The template does not have defaults set for all parameter values.
- The template fails AMS validation. For AMS validation steps, see the information later in this topic.

The RFC fails if the CloudFormation stack fails to create due to a resource creation issue.

To learn more about CFN validation and validator, see Template Validation and CloudFormation ingest stack: CFN validator examples.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-361tlo1k7339x.

Example: Required Parameters

```
{
  "StackId": "stack-kiwonebfnadq08sol",
  "VpcId": "vpc-01234567890abcdef",
  "TimeoutInMinutes": 360
}
```

Example: All Parameters

```
Example not available.
```

Directory Service Subcategory

Change Type Items and Operations in the Directory Service Subcategory

- Computer Object | Remove
- Computer Object | Remove SPN
- Directory | Accept Sharing
- Directory | Create AD Trust
- Directory | Share Directory
- Directory | Unshare Directory
- DNS | Add A Record
- DNS | Add CNAME Record
- DNS | Delete Conditional Forwarder
- DNS | Remove Record
- DNS | Update Cluster Permissions

- DNS | Update Conditional Forwarder
- DNS | Update Record Permission
- Users and Groups | Add Group
- Users and Groups | Add Group To Group
- Users and Groups | Add User To Group
- Users and Groups | Remove User from Group

Computer Object | Remove

Remove a stale computer object from Microsoft Active Directory (AD) and the corresponding DNS A and PTR records from DNS. Removing the computer object will prevent anyone from raising access against this host using the AMS access control. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Computer object | Remove

Change Type Details

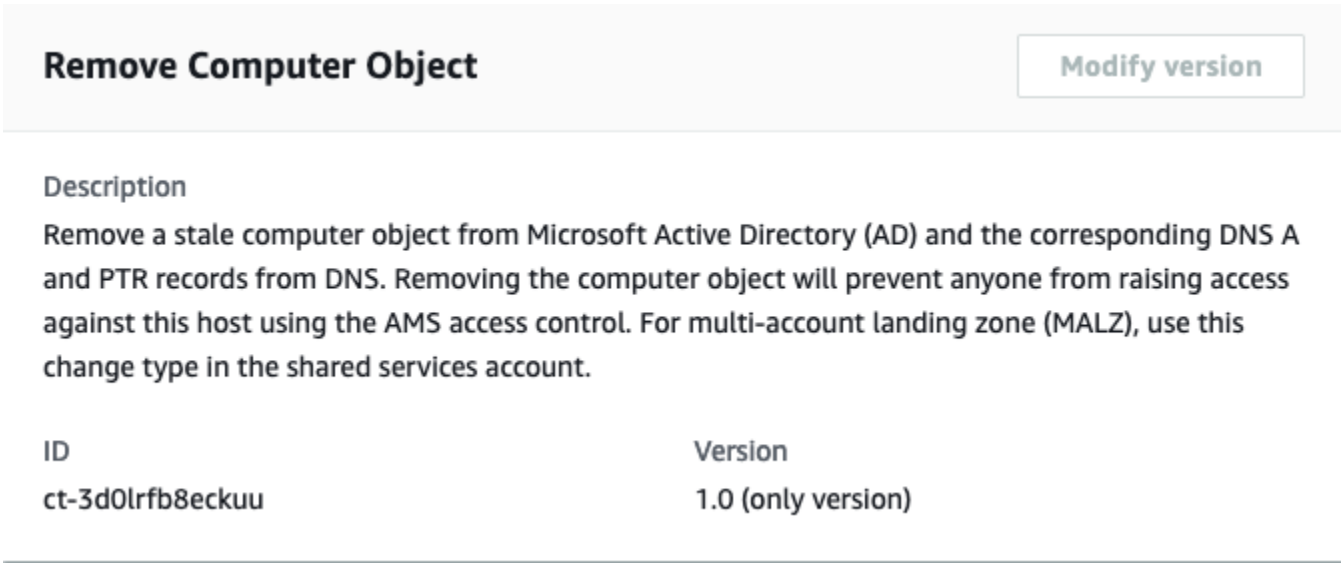
Change type ID	ct-3d0lrfb8eckuu
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove a computer object

Removing a computer object from an AMS-managed AD with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing a computer object from an AMS-managed AD with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3d0lrfb8eckuu" --change-type-version
 "1.0" --title "Remove Computer Object" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-RemoveADComputerObject-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"ADComputerName\": [\"ABRACADABRA\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ComputerObjectRemoveParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3d0lrfb8eckuu"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ComputerObjectRemoveParams.json
   ```

   Modify and save the ComputerObjectRemoveParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-RemoveADComputerObject-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "ADComputerName": [
         "ABRACADABRA"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it ComputerObjectRemoveRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ComputerObjectRemoveRfc.json
   ```

3. Modify and save the ComputerObjectRemoveRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-3d0lrfb8eckuu",
   ```

```
"ChangeTypeVersion":    "1.0",
"Title":                "Remove computer object"
}
```

4. Create the RFC, specifying the ComputerObjectRemoveRfc file and the ComputerObjectRemoveParams file:

```
aws amscm create-rfc --cli-input-json file://ComputerObjectRemoveRfc.json --
execution-parameters file://ComputerObjectRemoveParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3d0lrfb8eckuu.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-RemoveADComputerObject-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "Hostname" : [
      "ABRACADABRA"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-RemoveADComputerObject-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
```

```
    "Hostname" : [
      "ABRACADABRA"
    ]
  }
}
```

Computer Object | Remove SPN

Remove the Service Principal Name (SPN) associated with a specified hostname or host alias in Microsoft Active Directory. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Computer object | Remove SPN

Change Type Details

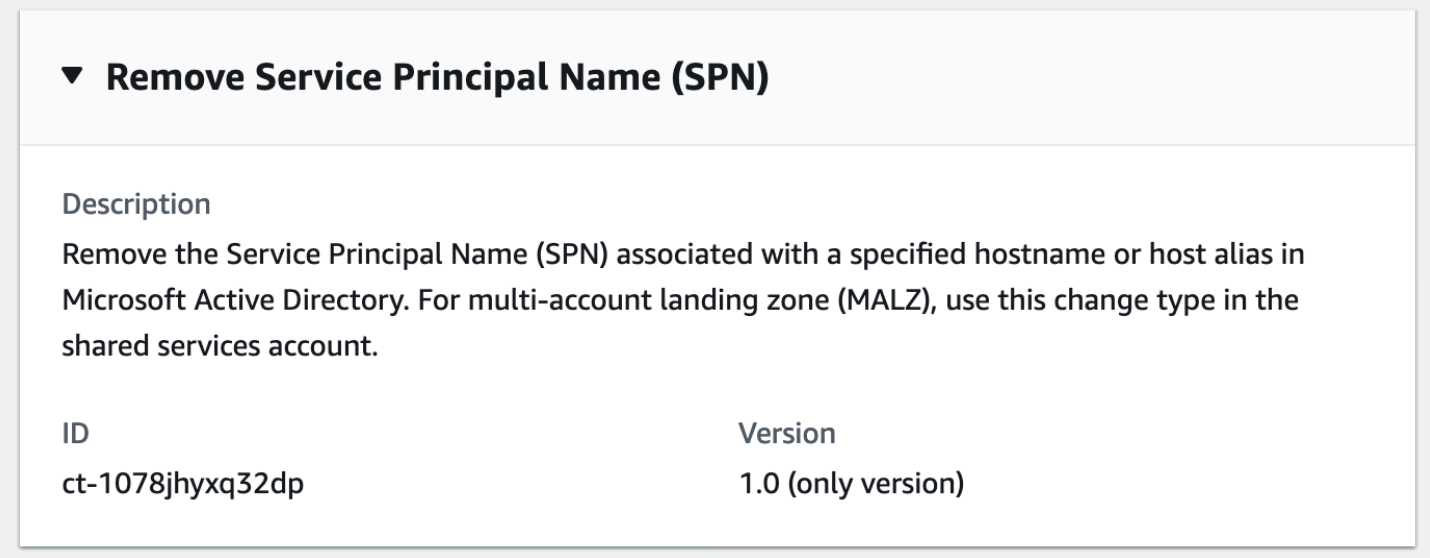
Change type ID	ct-1078jhyxq32dp
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove a computer object's SPN

Removing a computer object's SPN from an AMS-managed AD with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing a computer object's SPN from an AMS-managed AD with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1078jhyxq32dp" --change-type-version
 "1.0" --title "Remove AD Computer SPN" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-RemoveADComputerSPN-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"Hostname\": [\"webserver\"], \"ServiceType\": [\"HOST\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ComputerObjectRemoveSpnParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1078jhyxq32dp"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ComputerObjectRemoveSpnParams.json
   ```

   Modify and save the ComputerObjectRemoveSpnParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-RemoveADComputerSPN-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "Hostname": [
         "webserver"
       ],
       "ServiceType": [
         "HOST"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it ComputerObjectRemoveSpnRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ComputerObjectRemoveSpnRfc.json
   ```

3. Modify and save the ComputerObjectRemoveSpnRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1078jhyxq32dp",
  "Title": "Remove AD Computer SPN"
}
```

4. Create the RFC, specifying the ComputerObjectRemoveSpnRfc file and the ComputerObjectRemoveSpnParams file:

```
aws amscm create-rfc --cli-input-json file://ComputerObjectRemoveSpnRfc.json --
execution-parameters file://ComputerObjectRemoveSpnParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1078jhyxq32dp.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-RemoveADComputerSPN-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "Hostname": ["RDP-12345"],
    "ServiceType": ["HOST"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RemoveADComputerSPN-Admin",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "Hostname": ["RDP-12345"],
    "ServiceType": ["HOST"],
    "AliasName": ["Valid-Alias123"],
    "GroupManagedServiceAccountName": ["Valid-Name-456"],
    "Port": ["1122"]
  }
}
```

Directory | Accept Sharing

Accept a directory sharing request sent from the directory owner account. This is run in the directory consumer account.

Full classification: Management | Directory Service | Directory | Accept sharing

Change Type Details

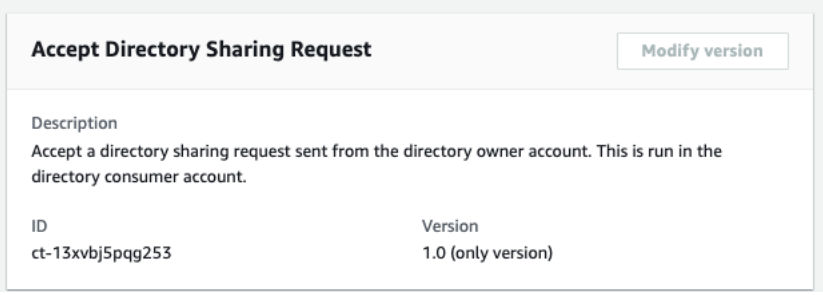
Change type ID	ct-13xvbj5pqg253
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Accept directory sharing request

Accept a directory sharing request with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Accept a directory sharing request with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \ --change-type-id "ct-13xvbj5pqg253" \ --change-type-version
 "1.0" --title "AWS Directory Service accept directory sharing" \ --execution-
parameters "{\"DocumentName\":\"AWSManagedServices-AcceptSharedDirectory\",\"Region
\":\"eu-central-1\",\"Parameters\":{\"SharedDirectoryId\":[\"d-000000000\"],
\"OwnerAccountId\":[\"000000000000\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DirectorySharingParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-13xvbj5pqg253"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DirectorySharingParams.json
```

   Modify and save the DirectorySharingParams file. For example, you can replace the contents with something like this:

```
{
{
  "DocumentName": "AWSManagedServices-AcceptSharedDirectory",
  "Region": "eu-central-1",
  "Parameters": {
    "SharedDirectoryId": ["d-000000000"],
    "OwnerAccountId": ["000000000000"]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it DirectorySharingRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DirectorySharingRfc.json
```

3. Modify and save the DirectorySharingRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":         "ct-13xvbj5pqg253",
"ChangeTypeVersion":    "1.0",
"Title":                "AWS Directory Service accept directory sharing"
}
```

4. Create the RFC, specifying the DirectorySharingRfc file and the DirectorySharingParams file:

```
aws amscm create-rfc --cli-input-json file://DirectorySharingRfc.json --execution-
parameters file://DirectorySharingParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type was originally classified as Management | Advanced stack components | Directory service | Accept sharing, and has now been moved to a more user friendly classification. The change type ID, ct-13xvbj5pqg253, has not changed.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-13xvbj5pqg253.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-AcceptSharedDirectory",
  "Region": "us-east-1",
  "Parameters": {
    "SharedDirectoryId": [
      "d-12e456789f"
    ],
    "OwnerAccountId": [
      "123456789012"
    ]
  }
}
```

Directory | Create AD Trust

Create a one-way trust between On-Prem Domain and (AWS) Managed Active Directory. For multi-account landing zone (MALZ), use this change type in the shared services account. Before creating

the trust, you need to make sure that the following prerequisites are met: 1. You must create the AD trust first on the On-Prem Domain and save the trust password in the Secrets Manager. 2. You must set up a Managed Active Directory (MAD) Security Group with an outbound rule that allows all traffic to On-Prem CIDR ranges.

Full classification: Management | Directory Service | Directory | Create AD trust

Change Type Details

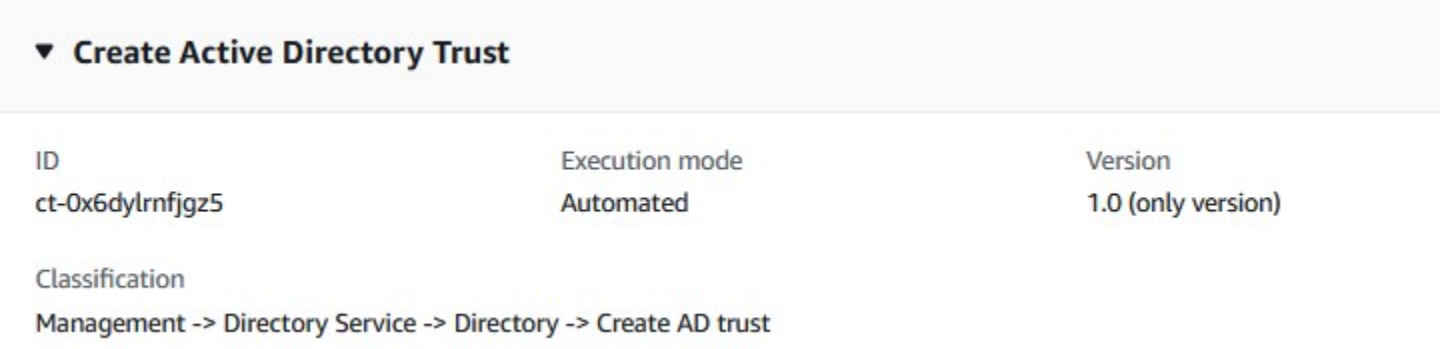
Change type ID	ct-0x6dylrnfjgz5
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Create Active Directory Trust

Adding an AD trust with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AD trust with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0x6dylrnfjgz5" --change-type-version "1.0" --
title "Create AD Trust" --execution-parameters '
{"DocumentName":"AWSManagedServices-CreateADTrust","Region":"ap-
southeast-2","Parameters":{"DirectoryId":["d-976774e42f"],"RemoteDomainName":
["onprem.local"],"SecretArn":["arn:aws:secretsmanager:ap-
southeast-2:996606605561:secret:customer-shared/CorrectTPW-BI79uu"],"TrustType":
["External"],"ConditionalForwarderIpAddresses":["10.153.28.39"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CreateADTrustParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0x6dylrnfjgz5" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CreateADTrustParams.json
   ```

Modify and save the CreateADTrustParams.json file. For example, you can replace the contents with something like this:

```
{
  "DocumentName": "AWSManagedServices-CreateADTrust",
  "Region": "ap-southeast-2",
  "Parameters": {
    "DirectoryId": [
      "d-976774e42f"
    ],
    "RemoteDomainName": [
      "onprem.local"
    ],
    "SecretArn": [
      "arn:aws:secretsmanager:ap-southeast-2:996606605561:secret:customer-shared/
CorrectTPW-BI79uu"
    ],
    "TrustType": [
      "External"
    ],
    "ConditionalForwarderIpAddresses": [
      "10.153.28.39"
    ]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it CreateADTrustRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateADTrustRfc.json
```

3. Modify and save the CreateADTrustRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-0x6dylrnfjgz5",
  "ChangeTypeVersion": "1.0",
  "Title": "Active Directory Trust"
}
```

4. Create the RFC, specifying the CreateADTrustRfc file and the CreateADTrustParams file:

```
aws amscm create-rfc --cli-input-json file://CreateADTrustRfc.json --execution-
parameters file://CreateADTrustParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0x6dylrnfjgz5.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-CreateADTrust",
    "Region": "us-east-1",
    "Parameters": {
        "DirectoryId": "d-12e456789f",
        "RemoteDomainName": "onprem.local",
        "SecretArn": "arn:aws:secretsmanager:us-east-1:000000000000:secret:customer-
shared/adtrust",
        "TrustType": "External",
        "ConditionalForwarderIpAddresses": "10.153.28.39,10.153.28.40"
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-CreateADTrust",
    "Region": "us-east-1",
    "Parameters": {
        "DirectoryId": "d-12e456789f",
        "RemoteDomainName": "onprem.local",
        "SecretArn": "arn:aws:secretsmanager:us-east-1:000000000000:secret:customer-
shared/adtrust",
```

```
        "TrustType": "External",
        "ConditionalForwarderIpAddresses": "10.153.28.39,10.153.28.40"
    }
}
```

Directory | Share Directory

Share a specified directory in your AWS account (directory owner) with another AWS account (directory consumer). Run this in your Shared Service account that has Managed Active Directory. This change type is only supported for multi-account landing zone (MALZ).

Full classification: Management | Directory Service | Directory | Share directory

Change Type Details

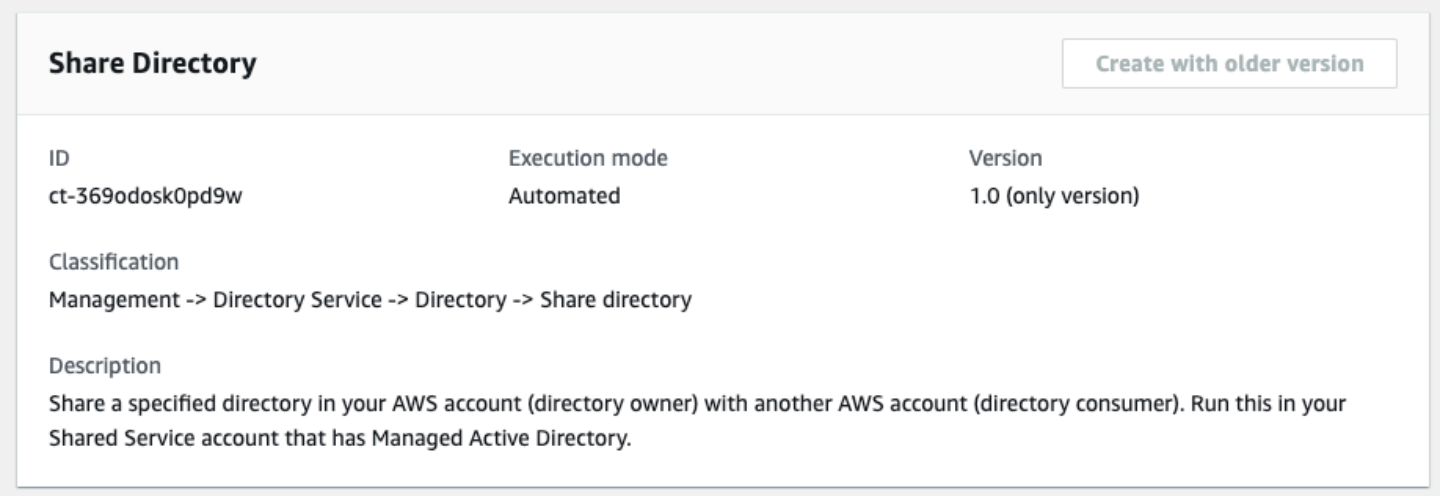
Change type ID	ct-369odosk0pd9w
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Share directory

Share a directory with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Share a directory with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-369odosk0pd9w" --change-type-version
 "1.0" --title "Share Directory" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ShareDirectory\",\"Region\":\"ap-southeast-2\",\"Parameters\":
{\"DirectoryId\":[\"d-123456ab7c\"],\"TargetAccountId\":[\"012345678912\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DirectorySharingParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-369odosk0pd9w"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DirectorySharingParams.json
   ```

   Modify and save the DirectorySharingParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-ShareDirectory",
     "Region": "us-east-1",
     "Parameters": {
       "DirectoryId": [
         "d-123456ab7c"
       ],
       "TargetAccountId": [
         "012345678912"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it DirectorySharingRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DirectorySharingRfc.json
   ```

3. Modify and save the DirectorySharingRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeId": "ct-369odosk0pd9w",
     "ChangeTypeVersion": "1.0",
   ```

```
  "Title": "Share Directory"
}
```

4. Create the RFC, specifying the DirectorySharingRfc file and the DirectorySharingParams file:

```
aws amscm create-rfc --cli-input-json file://DirectorySharingRfc.json --execution-
parameters file://DirectorySharingParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For related CTs, see Directory Service Subcategory.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-369odosk0pd9w.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-ShareDirectory",
  "Region": "us-east-1",
  "Parameters": {
    "DirectoryId": [
      "d-0000000000"
    ],
    "TargetAccountId": [
      "000000000000"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ShareDirectory",
  "Region": "us-east-1",
  "Parameters": {
```

```
    "DirectoryId": [
      "d-12e456789f"
    ],
    "TargetAccountId": [
      "123456789012"
    ]
  }
}
```

Directory | Unshare Directory

Stops the directory sharing between the directory owner and consumer accounts. Run this in your Shared Service account that has Managed Active Directory. This change type is only supported for multi-account landing zone (MALZ).

Full classification: Management | Directory Service | Directory | Unshare directory

Change Type Details

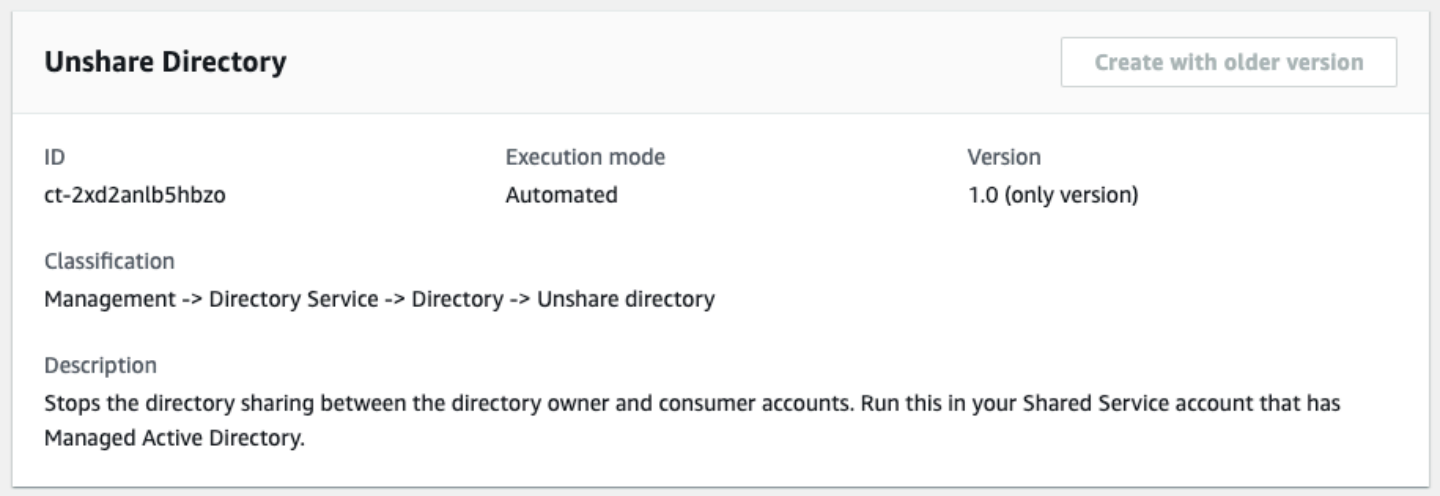
Change type ID	ct-2xd2anlb5hbzo
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Unshare directory

Unshare a directory with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Unshare a directory with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm  create-rfc --change-type-id "ct-2xd2anlb5hbzo" --change-type-version
 "1.0" --title "Unshare Directory" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-ShareDirectory\",\"Region\":\"ap-southeast-2\",\"Parameters\":
{\"DirectoryId\":[\"d-123456ab7c\"],\"UnshareTarget\":[\"012345678912\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DirectoryUnsharingParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2xd2anlb5hbzo"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DirectoryUnsharingParams.json
   ```

   Modify and save the DirectoryUnsharingParams.json file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UnshareDirectory",
     "Region": "us-east-1",
     "Parameters": {
       "DirectoryId": [
         "d-123456ab7c"
       ],
       "UnshareTarget": [
         "012345678912"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it DirectoryUnsharingRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DirectoryUnsharingRfc.json
   ```

3. Modify and save the DirectoryUnsharingRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeId": "ct-2xd2anlb5hbzo",
     "ChangeTypeVersion": "1.0",
   ```

```
  "Title": "Unshare Directory"
}
```

4. Create the RFC, specifying the DirectoryUnsharingRfc file and the DirectoryUnsharingParams file:

```
aws amscm create-rfc --cli-input-json file://DirectoryUnsharingRfc.json --
execution-parameters file://DirectoryUnsharingParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For related CTs, see Directory Service Subcategory.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2xd2anlb5hbzo.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UnshareDirectory",
  "Region": "us-east-1",
  "Parameters": {
    "DirectoryId": [
      "d-0000000000"
    ],
    "UnshareTarget": [
      "000000000000"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UnshareDirectory",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "DirectoryId": [
      "d-12e456789f"
    ],
    "UnshareTarget": [
      "123456789012"
    ]
  }
}
```

DNS | Add A Record

Add a new static DNS A record in AWS Managed Microsoft Active Directory (AD). For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Add A record

Change Type Details

Change type ID	ct-2w3rbmnny1qpo
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add DNS "A" record

Adding a DNS A record with the console

The following shows this change type in the AMS console.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a DNS A record with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2w3rbmnny1qpo" --change-type-version
 "1.0" --title "Add DNS A Record" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-CreateDNSARecord-Admin\",\"Region\": \"us-east-1\",\"Parameters
\": {\"RecordName\": [\"web-server\"], \"IPAddress\": [\"132.133.134.135\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ArecordAddParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2w3rbmnny1qpo" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > ArecordAddParams.json
   ```

   Modify and save the ArecordAddParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateDNSARecord-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RecordName": [
         "web-server"
       ],
       "IPAddress": [
         "132.133.134.135"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it ArecordAddRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ArecordAddRfc.json
   ```

3. Modify and save the ArecordAddRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":          "ct-2w3rbmnny1qpo",
   "ChangeTypeVersion":     "1.0",
   "Title":                 "AWS Directory Service add DNS A record"
   ```

```
}
```

4. Create the RFC, specifying the ArecordAddRfc file and the ArecordAddParams file:

```
aws amscm create-rfc --cli-input-json file://ArecordAddRfc.json --execution-
parameters file://ArecordAddParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> For multi-account landing zone (MALZ), use this change type in the shared services account.

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2w3rbmnny1qpo.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDNSARecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordName": ["web-server"],
    "IPAddress": ["123.1.2.3"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDNSARecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
```

```
    "RecordName": ["web-server"],
    "IPAddress": ["123.1.2.3"],
    "TTLValue": ["01:00:01"]
  }
}
```

DNS | Add CNAME Record

Create a new DNS CNAME record in AWS Managed Microsoft Active Directory (AD). CNAME records must always point to another domain name, never directly to an IP address. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Add CNAME record

Change Type Details

Change type ID	ct-2murl5xzbxoxf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add DNS CNAME record in AMS

> **Note**
>
> To create a CNAME record in AWS, see How do I create alias records for services hosted in AWS?.

Adding a DNS CNAME record with the console

The following shows this change type in the AMS console.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a DNS CNAME record with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2murl5xzbxoxf" --change-type-version
 "1.0" --title "Add DNS CNAME Record" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-CreateDNSCnameRecord-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"RecordName\": [\"host1.mycompany.com\"], \"RecordCname\": [\"web-
server\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CnameRecordAddParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2w3rbmnny1qpo" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > CnameRecordAddParams.json
   ```

   Modify and save the CnameRecordAddParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-CreateDNSCnameRecord-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RecordName": [
         "host1.mycompany.com"
       ],
       "RecordCname": [
         "web-server"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it CnameRecordAddRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CnameRecordAddRfc.json
   ```

3. Modify and save the CnameRecordAddRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":          "ct-2murl5xzbxoxf",
   "ChangeTypeVersion":     "1.0",
   ```

```
"Title":                "AWS Directory Service add DNS CNAME record"
}
```

4. Create the RFC, specifying the CnameRecordAddRfc file and the CnameRecordAddParams file:

```
aws amscm create-rfc --cli-input-json file://CnameRecordAddRfc.json --execution-
parameters file://CnameRecordAddParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For multi-account landing zone (MALZ), use this change type in the Shared Services account.
- For information about Directory Service, see the Directory Service Admin Guide. To learn about CNAME records, see CNAME record type.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2murl5xzbxoxf.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDNSCnameRecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordName": ["hostname123.example.com"],
    "RecordCname": ["webserver"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateDNSCnameRecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
```

```
    "RecordName": ["hostname123.example.com"],
    "RecordCname": ["webserver"]
  }
}
```

DNS | Delete Conditional Forwarder

Delete AD DNS conditional forwarder for a remote domain. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Delete conditional forwarder

Change Type Details

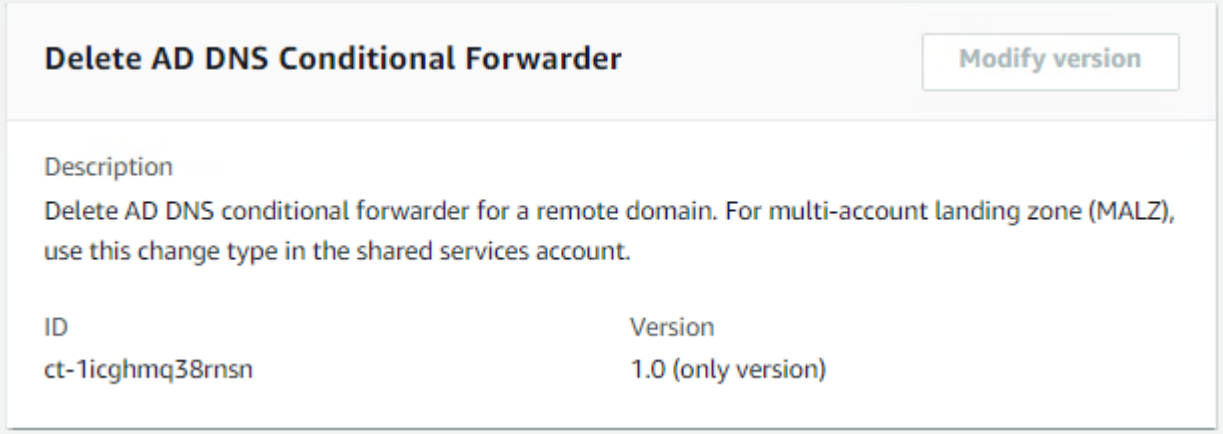
Change type ID	ct-1icghmq38rnsn
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete a DNS conditional forwarder

Deleting DNS conditional forwarders with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting DNS conditional forwarders with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1icghmq38rnsn" --change-type-version
 "1.0" --title "AWSManagedServices-DeleteADDNSConditionalForwarder" --execution-
parameters "{\"DocumentName\": \"AWSManagedServices-DeleteADDNSConditionalForwarder-
Admin\",\"Region\": \"us-east-1\",\"Parameters\": {\"RemoteDomainName\":
 [\"test.forwarders.com\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CondForwardDeleteParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1icghmq38rnsn"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CondForwardDeleteParams.json
   ```

   Modify and save the CondForwardDeleteParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DeleteADDNSConditionalForwarder-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RemoteDomainName": [
         "test.forwarders.com"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it CondForwardDeleteRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > CondForwardDeleteRfc.json
   ```

3. Modify and save the CondForwardDeleteRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-1icghmq38rnsn",
     "Title": "Delete AD DNS Conditional Forwarder"
   ```

```
}
```

4. Create the RFC, specifying the CondForwardDeleteRfc file and the CondForwardDeleteParams file:

```
aws amscm create-rfc --cli-input-json file://CondForwardDeleteRfc.json --execution-
parameters file://CondForwardDeleteParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1icghmq38rnsn.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-DeleteADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"]
  }
}
```

DNS | Remove Record

Remove the specified DNS resource record name, either an A or CNAME, or pointer record (PTR), from the specified DNS zone. By default, only the static record is removed per specified RecordName for A or CNAME records. Use the RecordData parameter to remove duplicates if there are multiple records with the same Host Name (RecordType A), either dynamic or static. For a PTR record type, all the static and dynamic records will be removed. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Remove record

Change Type Details

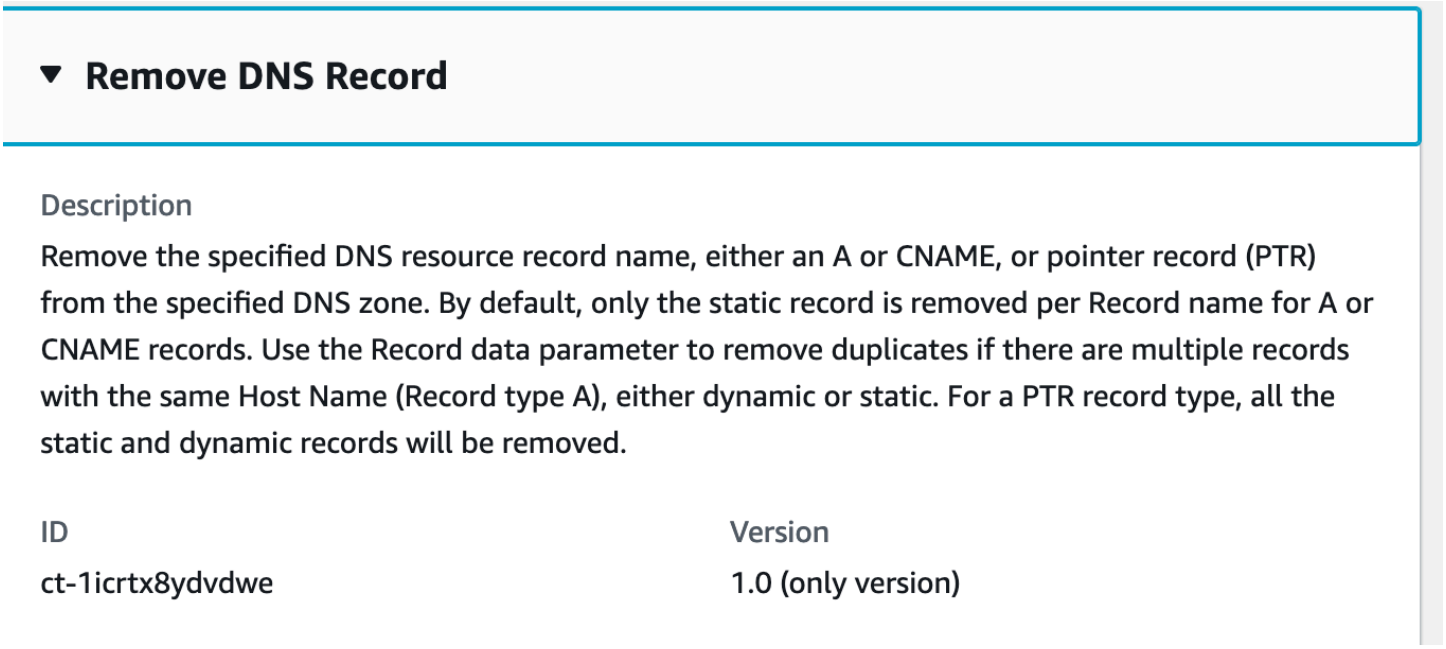
Change type ID	ct-1icrtx8ydvdwe
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove record

Removing a DNS record with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing a DNS record with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1icrtx8ydvdwe" --change-type-version
 "1.0" --title "Remove DNS Record" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-RemoveDNSRecord-Admin\",\"Region\": \"us-east-1\",\"Parameters\":
 {\"RecordName\": [\"web-server\"], \"RecordType\": [\"CNAME\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RecordRemoveParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1icrtx8ydvdwe" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RecordRemoveParams.json
   ```

   Modify and save the RecordRemoveParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-RemoveDNSRecord-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RecordName": [
         "web-server"
       ],
       "RecordType": [
         "CNAME"
       ]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it RecordRemoveRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RecordRemoveRfc.json
   ```

3. Modify and save the RecordRemoveRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-1icrtx8ydvdwe",
"ChangeTypeVersion":     "1.0",
"Title":                 "Remove DNS record"
}
```

4. Create the RFC, specifying the RecordRemoveRfc file and the RecordRemoveParams file:

```
aws amscm create-rfc --cli-input-json file://RecordRemoveRfc.json --execution-
parameters file://RecordRemoveParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1icrtx8ydvdwe.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-RemoveDNSRecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordName": ["123.123.123.123"],
    "RecordType": ["PTR"]
  }
}
```

Example: All Parameters

```
{
```

```
  "DocumentName": "AWSManagedServices-RemoveDNSRecord-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordName": ["web-server"],
    "RecordType": ["CNAME"],
    "RecordData": ["123.123.123.123"]
  }
}
```

DNS | Update Cluster Permissions

Grants full control to the Cluster object on the Listener object to bring the SQL Server Listener object online. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Update cluster permissions

Change Type Details

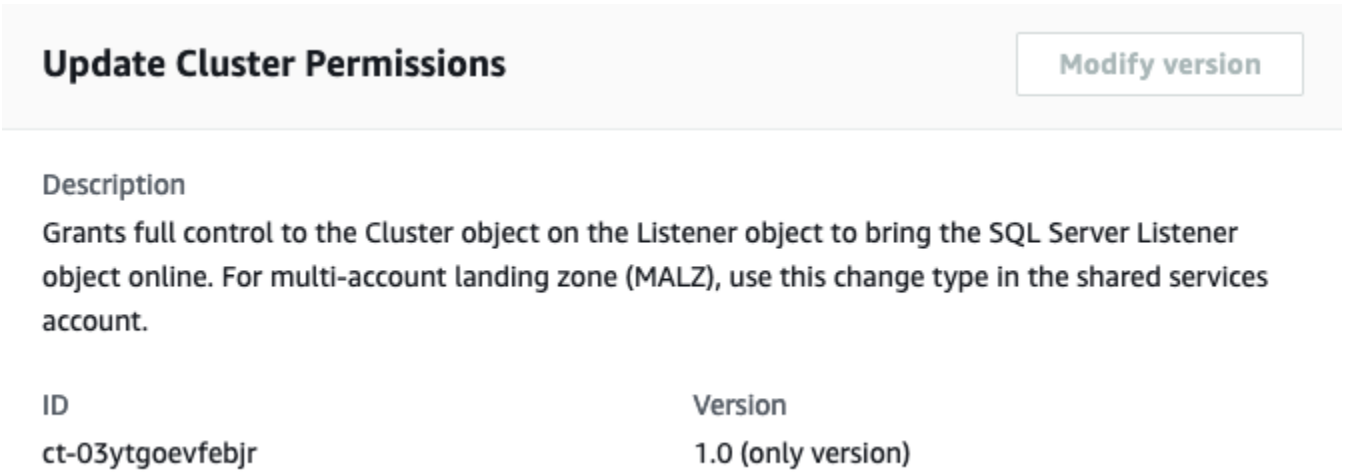
Change type ID	ct-03ytgoevfebjr
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update cluster permissions

Updating cluster permissions with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating cluster permissions with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-03ytgoevfebjr" --change-type-version "1.0"
 --title "Update Cluster Permissions" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-UpdateClusterDNSPermission-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"ClusterName\": [\"EC2-SAMPLE-AGL\"],\"ClusterNodeComputerName\":
[\"EC2SAMPLE-01A1MR9\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ClusterPermissionsUpdateParams.json:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateClusterDNSPermission-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "ClusterName": ["EC2-SAMPLE-AGL"],
       "ClusterNodeComputerName": ["EC2SAMPLE-01A1MR9"]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it ClusterPermissionsUpdateRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > ClusterPermissionsUpdateRfc.json
   ```

3. Modify and save the ClusterPermissionsUpdateRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":          "ct-03ytgoevfebjr",
   "ChangeTypeVersion":     "1.0",
   "Title":                 "Update Cluster Permissions"
   }
   ```

4. Create the RFC, specifying the ClusterPermissionsUpdateRfc file and the ClusterPermissionsUpdateParams file:

   ```
   aws amscm create-rfc --cli-input-json file://ClusterPermissionsUpdateRfc.json --
   execution-parameters file://ClusterPermissionsUpdateParams.json
   ```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For additional information, see DirectoryService section.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-03ytgoevfebjr.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateClusterDNSPermission-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "ClusterName": ["EC2-O6G85G-AGL"],
    "ClusterNodeComputerName": ["EC2AMAZ-O6G3MR9"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateClusterDNSPermission-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "ClusterName": ["EC2-O6G85G-AGL"],
    "ClusterNodeComputerName": ["EC2AMAZ-O6G3MR9"]
  }
}
```

DNS | Update Conditional Forwarder

Update AD DNS conditional forwarder for a remote domain. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Update conditional forwarder

Change Type Details

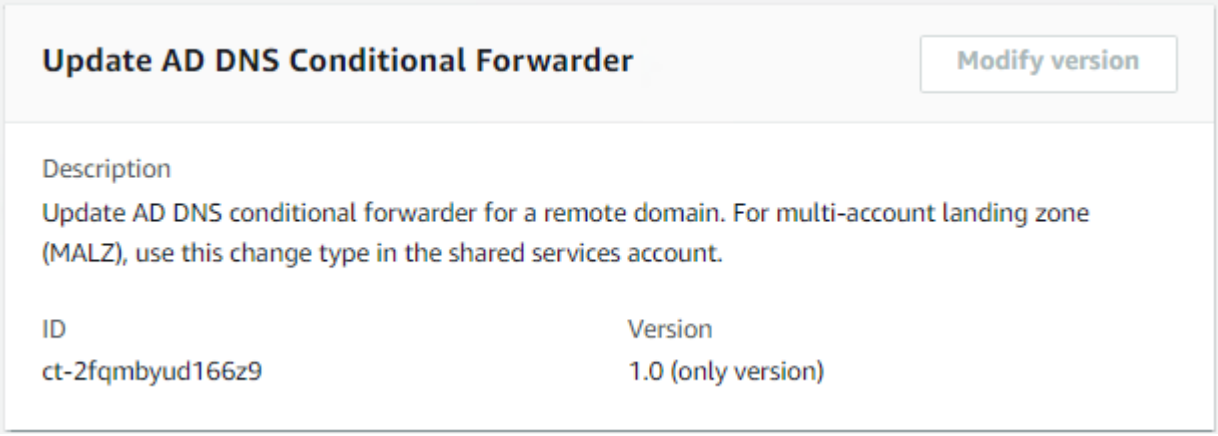
Change type ID	ct-2fqmbyud166z9
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update a DNS conditional forwarder

Updating DNS conditional forwarders with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating DNS conditional forwarders with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2fqmbyud166z9" --change-type-version
 "1.0" --title "AWSManagedServices-UpdateADDNSConditionalForwarder" --execution-
parameters "{\"DocumentName\": \"AWSManagedServices-UpdateADDNSConditionalForwarder-
Admin\",\"Region\": \"us-east-1\",\"Parameters\": {\"RemoteDomainName\":
 [\"test.forwarders.com\"], \"IpAddresses\": [\"10.0.0.3\", \"10.0.0.4"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it CondForwardUpdateParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2fqmbyud166z9"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CondForwardUpdateParams.json
   ```

   Modify and save the CondForwardUpdateParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-UpdateADDNSConditionalForwarder-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RemoteDomainName": [
      "Domain_Name"
    ],
    "IPAddresses": [
      "132.133.134.135", "135.134.133.132"
    ]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it CondForwardUpdateRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CondForwardUpdateRfc.json
```

3. Modify and save the CondForwardUpdateRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-2fqmbyud166z9",
"ChangeTypeVersion":     "1.0",
"Title":                 "Update conditional forwarders"
}
```

4. Create the RFC, specifying the CondForwardUpdateRfc file and the CondForwardUpdateParams file:

```
aws amscm create-rfc --cli-input-json file://CondForwardUpdateRfc.json --execution-
parameters file://CondForwardUpdateParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2fqmbyud166z9.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-UpdateADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"],
    "IPAddresses": ["10.0.0.1", "10.0.0.2"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-UpdateADDNSConditionalForwarder-Admin",
  "Region" : "us-east-1",
  "Parameters": {
    "RemoteDomainName": ["test.test1.com"],
    "IPAddresses": ["10.0.0.1", "10.0.0.2"]
  }
}
```

DNS | Update Record Permission

Grant permissions to the computer object to update DNS records after failover. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | DNS | Update record permission

Change Type Details

Change type ID	ct-1eft8s6vdhz0w
Current version	1.0
Expected execution duration	60 minutes

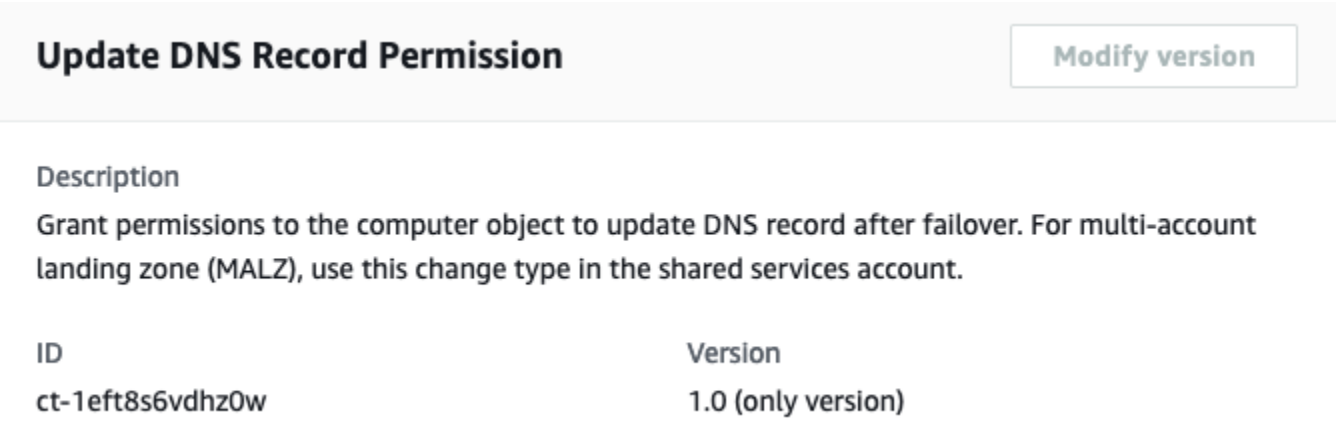
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update DNS record permission

Updating DNS record permissions with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating DNS record permissions with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1eft8s6vdhz0w" --change-type-version
 "1.0" --title "Update DNS Record" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-UpdateDNSRecordsPermission-Admin\",\"Region\": \"us-east-1\",
\"Parameters\": {\"RecordNames\": [\"EC2CLUS-SAMPLE\",\"EC2SAmPL1-AWS\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateDNSRecordsPermissionParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1eft8s6vdhz0w"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateDNSRecordsPermissionParams.json
   ```

   Modify and save the UpdateDNSRecordsPermissionParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateDNSRecordsPermission-Admin",
     "Region": "us-east-1",
     "Parameters": {
       "RecordNames": ["EC2CLUS-SAMPLE","EC2SAmPL1-AWS"]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it UpdateDNSRecordsPermissionRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateDNSRecordsPermissionRfc.json
```

3. Modify and save the UpdateDNSRecordsPermissionRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1eft8s6vdhz0w",
  "Title": "Update DNS record"
}
```

4. Create the RFC, specifying the UpdateDNSRecordsPermissionRfc file and the UpdateDNSRecordsPermissionParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateDNSRecordsPermissionRfc.json --
execution-parameters file://UpdateDNSRecordsPermissionParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1eft8s6vdhz0w.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateDNSRecordsPermission-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordNames": ["EC2CLUS-SAMPl1,EC2G90BI1-AWS"]
  }
}
```

Example: All Parameters

```
{
```

```
  "DocumentName": "AWSManagedServices-UpdateDNSRecordsPermission-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "RecordNames": ["EC2CLUS-SAMPl1,EC2G90BI1-AWS"]
  }
}
```

Users and Groups | Add Group

Create an Active Directory (AD) group in the AMS managed AD. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Users and groups | Add group

Change Type Details

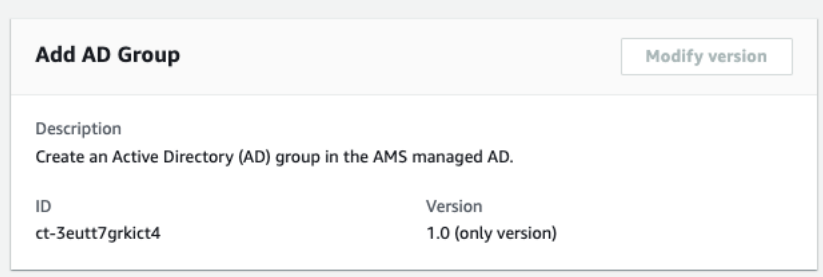
Change type ID	ct-3eutt7grkict4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add Active Directory group

Adding an AD group with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AD group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3eutt7grkict4" --change-type-version
 "1.0" --title "Create AD group" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CreateADGroup-Admin\",\"Region\":\"us-east-1\",\"Parameters\":
{\"GroupName\":[\"my-group\"],\"GroupDescription\":[\"Group description\"],\"GroupScope
\":[\"DomainLocal\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AdGroupAddParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-3eutt7grkict4" --query
 "ChangeTypeVersion.ExecutionInputSchema" --output text > AdGroupAddParams.json
```

   Modify and save the AdGroupAddParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName" : "AWSManagedServices-CreateADGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "GroupName" : ["my-group"],
    "GroupDescription" : ["Group description"],
    "GroupScope" : ["DomainLocal"]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it AdGroupAddRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AdGroupAddRfc.json
```

3. Modify and save the AdGroupAddRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3eutt7grkict4",
  "Title": "Create AD group"
}
```

4. Create the RFC, specifying the AdGroupAddRfc file and the AdGroupAddParams file:

```
aws amscm create-rfc --cli-input-json file://AdGroupAddRfc.json --execution-
parameters file://AdGroupAddParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3eutt7grkict4.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateADGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "GroupName" : ["my-group"],
    "GroupDescription" : ["Group description"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-CreateADGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "GroupName" : ["my-group"],
    "GroupDescription" : ["Group description"],
    "GroupScope" : ["DomainLocal"]
  }
}
```

Users and Groups | Add Group To Group

Add an Active Directory (AD) group in the trusted domain to an AD group in the AMS managed AD. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Users and groups | Add group to group

Change Type Details

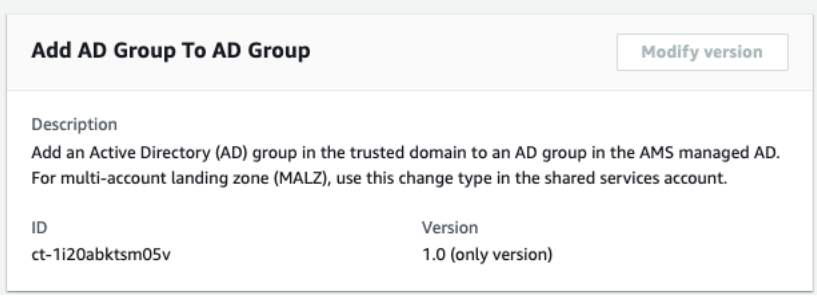
Change type ID	ct-1i20abktsm05v
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add an AD group to an AD group

Adding an AD group to an AMS-managed AD group with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AD group to an AMS-managed AD group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1i20abktsm05v" --change-type-version
 "1.0" --title "Add AD group to AD group" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-AddADGroupToADGroup-Admin\",\"Region\":\"us-east-1\",\"Parameters
\":{\"NestedGroupName\":[\"my-nested-group\"],\"GroupName\":[\"my-parent-group\"],
\"TrustedDomainFQDN\":[\"my-domain.com\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it GroupToGroupAddParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1i20abktsm05v"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    GroupToGroupAddParams.json
   ```

   Modify and save the GroupToGroupAddParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-AddADGroupToADGroup-Admin",
     "Region" : "us-east-1",
     "Parameters" : {
       "NestedGroupName" : ["my-nested-group"],
   ```

```
    "GroupName" : ["my-parent-group"],
    "TrustedDomainFQDN" : ["my-domain.com"]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it GroupToGroupAddRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > GroupToGroupAddRfc.json
```

3. Modify and save the GroupToGroupAddRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-1i20abktsm05v",
"ChangeTypeVersion":     "1.0",
"Title":                 "Add AD group to AD group"
}
```

4. Create the RFC, specifying the GroupToGroupAddRfc file and the GroupToGroupAddParams file:

```
aws amscm create-rfc --cli-input-json file://GroupToGroupAddRfc.json --execution-
parameters file://GroupToGroupAddParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1i20abktsm05v.

Example: Required Parameters

```
{
```

```
  "DocumentName" : "AWSManagedServices-AddADGroupToADGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "NestedGroupName" : ["nested-group"],
    "GroupName" : ["parent-group"],
    "TrustedDomainFQDN" : ["my-test-domain.com"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddADGroupToADGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "NestedGroupName" : ["nested-group"],
    "GroupName" : ["parent-group"],
    "TrustedDomainFQDN" : ["my-test-domain.com"]
  }
}
```

Users and Groups | Add User To Group

Add an Active Directory (AD) user to an AD group in the AMS managed AD. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Users and groups | Add user to group

Change Type Details

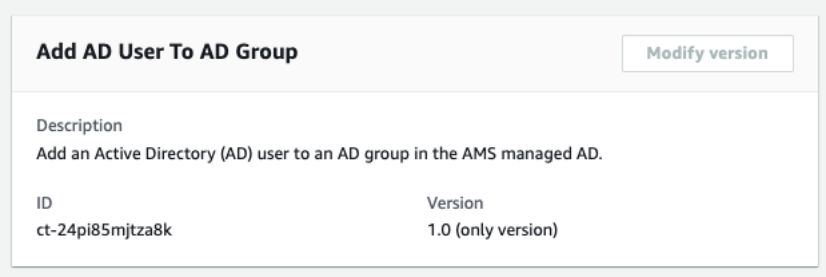
Change type ID	ct-24pi85mjtza8k
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add an AD user to an AD group

Adding an AD user to an AMS-managed AD group with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding an AD user to an AMS-managed AD group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-24pi85mjtza8k" --change-type-version
 "1.0" --title "Add AD user to AD group" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-AddADUserToGroup-Admin\",\"Region\":\"us-east-1\",\"Parameters
\":{\"UserName\":[\"my-user\"],\"GroupName\":[\"my-group\"],\"DomainFQDN\":[\"my-
domain.com\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UserToGroupAddParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-24pi85mjtza8k" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UserToGroupAddParams.json
   ```

   Modify and save the UserToGroupAddParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName" : "AWSManagedServices-AddADUserToGroup-Admin",
     "Region" : "us-east-1",
     "Parameters" : {
       "UserName" : ["my-user"],
       "GroupName" : ["my-group"],
       "DomainFQDN" : ["my-domain.com"]
     }
   }
   ```

2. Output the RFC template to a file in your current folder; this example names it UserToGroupAddRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UserToGroupAddRfc.json
   ```

3. Modify and save the UserToGroupAddRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":         "ct-24pi85mjtza8k",
   ```

```
"ChangeTypeVersion":    "1.0",
"Title":                "Add AD user to AD group"
}
```

4. Create the RFC, specifying the UserToGroupAddRfc file and the UserToGroupAddParams file:

```
aws amscm create-rfc --cli-input-json file://UserToGroupAddRfc.json --execution-
parameters file://UserToGroupAddParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-24pi85mjtza8k.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddADUserToGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "UserName" : ["my-user"],
    "GroupName" : ["parent-group"],
    "DomainFQDN" : ["my-test-domain.com"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-AddADUserToGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "UserName" : ["my-user"],
    "GroupName" : ["parent-group"],
```

```
    "DomainFQDN" : ["my-test-domain.com"]
  }
}
```

Users and Groups | Remove User from Group

Remove an Active Directory (AD) user from an AD group in the AMS managed AD. For multi-account landing zone (MALZ), use this change type in the shared services account.

Full classification: Management | Directory Service | Users and groups | Remove user from group

Change Type Details

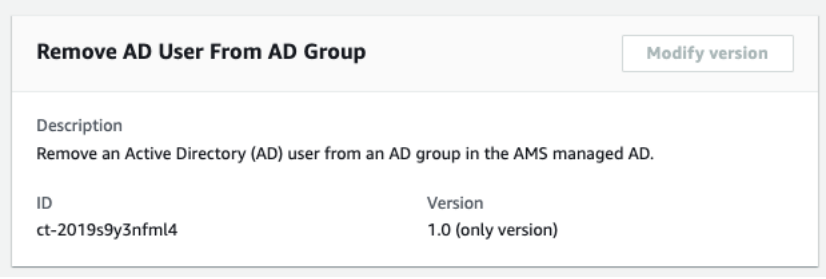
Change type ID	ct-2019s9y3nfml4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove an AD user from an AD group

Removing an AD user from an AMS-managed AD group with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing an AD user from an AMS-managed AD group with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2019s9y3nfml4" --change-type-version "1.0"
 --title "Remove AD user from AD group" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-RemoveADUserFromGroup-Admin\",\"Region\":\"us-east-1\",
\"Parameters\":{\"UserName\":[\"my-user\"],\"GroupName\":[\"my-group\"],\"DomainFQDN\":
[\"my-domain.com\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UserFromGroupRemoveParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2019s9y3nfml4"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UserFromGroupRemoveParams.json
   ```

Modify and save the UserFromGroupRemoveParams file. For example, you can replace the contents with something like this:

```
{
  "DocumentName" : "AWSManagedServices-RemoveADUserFromGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "UserName" : ["my-user"],
    "GroupName" : ["my-group"],
    "DomainFQDN" : ["my-domain.com"]
  }
}
```

2. Output the RFC template to a file in your current folder; this example names it UserFromGroupRemoveRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UserFromGroupRemoveRfc.json
```

3. Modify and save the UserFromGroupRemoveRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-2019s9y3nfml4",
"ChangeTypeVersion":     "1.0",
"Title":                 "Remove AD user from AD group"
}
```

4. Create the RFC, specifying the UserFromGroupRemoveRfc file and the UserFromGroupRemoveParams file:

```
aws amscm create-rfc --cli-input-json file://UserFromGroupRemoveRfc.json --
execution-parameters file://UserFromGroupRemoveParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Directory Service, see the Directory Service Admin Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2019s9y3nfml4.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-RemoveADUserFromGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "UserName" : ["my-user"],
    "GroupName" : ["my-group"],
    "DomainFQDN" : ["my-domain.com"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-RemoveADUserFromGroup-Admin",
  "Region" : "us-east-1",
  "Parameters" : {
    "UserName" : ["my-user"],
    "GroupName" : ["my-group"],
    "DomainFQDN" : ["my-domain.com"]
  }
}
```

Host Security Subcategory

Change Type Items and Operations in the Host Security Subcategory

- Malware Full System Scan | Disable (Review Required)
- Trend Micro DSM | Add Login (Read-Only)

Malware Full System Scan | Disable (Review Required)

Use to disable periodic malware full system scan feature in all EC2 instances deployed in a single VPC.

Full classification: Management | Host security | Malware full system scan | Disable (review required)

Change Type Details

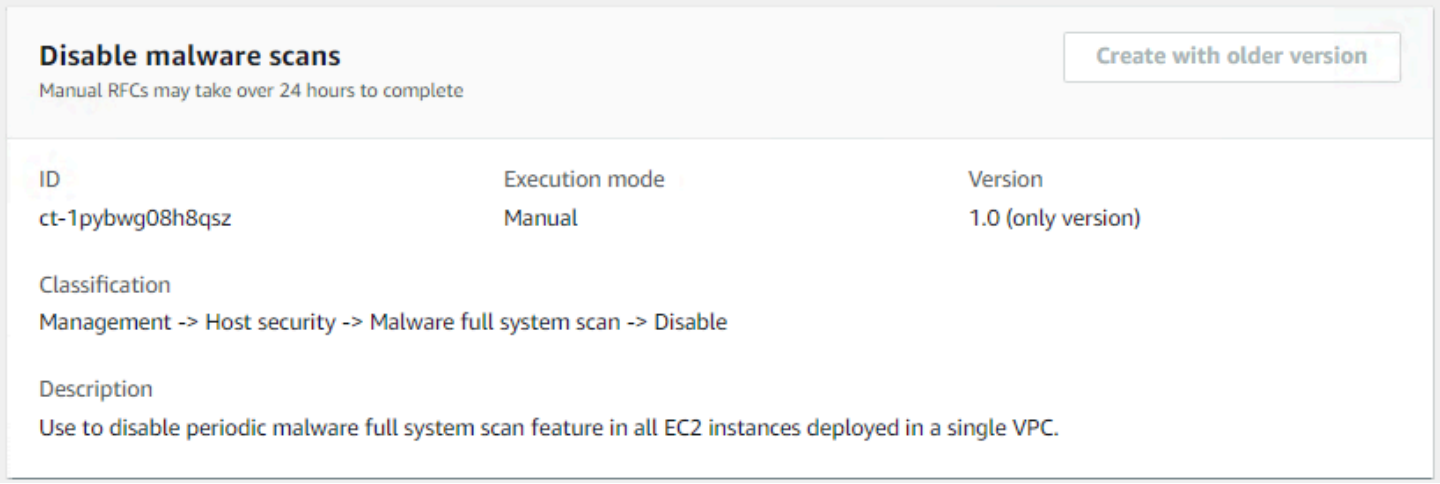
Change type ID	ct-1pybwg08h8qsz
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Disable malware scanning on a VPC

Disabling malware scanning with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Disabling malware scanning with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1pybwg08h8qsz" --change-type-version "1.0"
 --title "Disable malware scanning" --execution-parameters "{\"VpcId\": \"VPC-ID\",
\"Priority\": \"High\}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it DisableMalwareScanParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1pybwg08h8qsz"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DisableMalwareScanParams.json
   ```

2. Modify and save the DisableMalwareScanParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-TerminateStandaloneInstances",
   ```

```
  "Region": "us-east-1",
  "Confirmation": "terminate instances",
  "Parameters": {
    "InstanceIds": [
      "i-1234567890abcdef0"
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it DisableMalwareScanRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DisableMalwareScanRfc.json
```

4. Modify and save the DisableMalwareScanRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "cct-1pybwg08h8qsz",
    "ChangeTypeVersion": "1.0",
    "Title": "Disable malware scanning"
}
```

5. Create the RFC, specifying the DisableMalwareScanRfc file and the DisableMalwareScanParams file:

```
aws amscm create-rfc --cli-input-json file://DisableMalwareScanRfc.json  --
execution-parameters file://DisableMalwareScanParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1pybwg08h8qsz.

Example: Required Parameters

```
{
  "VpcId": "vpc-28abd91e"
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-28abd91e",
  "Priority": "Medium"
}
```

Trend Micro DSM | Add Login (Read-Only)

Request a read-only login to the Trend Micro console for your account.

Full classification: Management | Host security | Trend Micro DSM | Add login (read-only)

Change Type Details

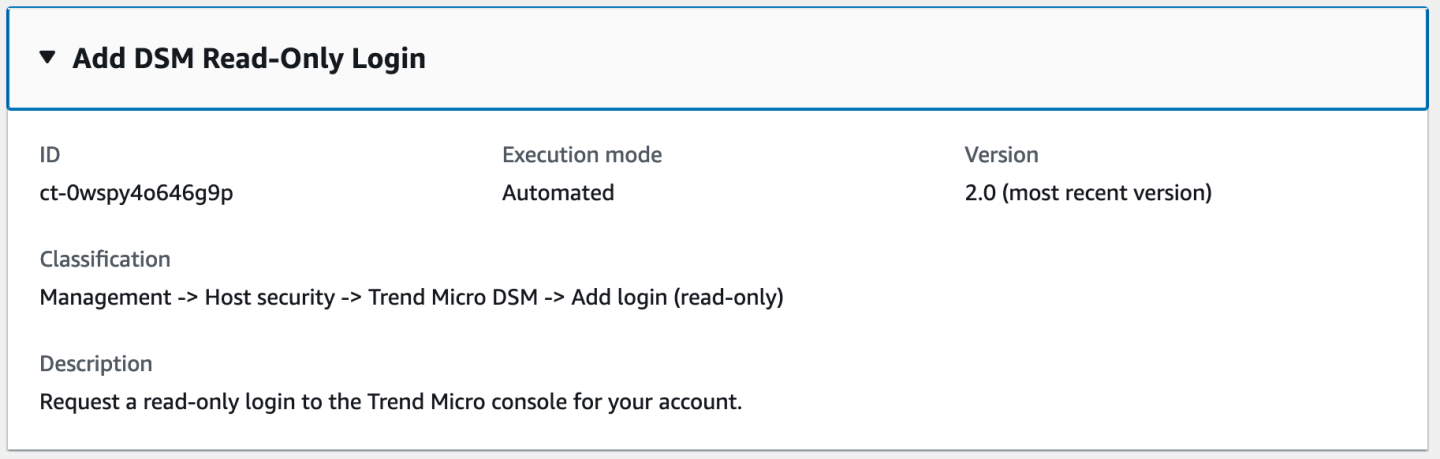
Change type ID	ct-0wspy4o646g9p
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add Trend Micro login (read only)

Adding a Trend Micro login (read only) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding a Trend Micro login (read only) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0wspy4o646g9p" --change-type-version "2.0"
 --title "Trend Micro Console Access" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-CreateEPSDSMReadOnlyUser\",\"Region\":\"us-east-1\",\"Parameters
\":{\"Username\":[\"eps-dsm-read-only-user\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it AddTrendMicroLoginParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0wspy4o646g9p"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AddTrendMicroLoginParams.json
   ```

2. Modify and save the AddTrendMicroLoginParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-CreateEPSDSMReadOnlyUser",
       "Region": "us-east-1",
       "Parameters": {
           "Username": [
               "eps-dsm-read-only-user"
           ]
       }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it AddTrendMicroLoginRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AddTrendMicroLoginRfc.json
   ```

4. Modify and save the AddTrendMicroLoginRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-0wspy4o646g9p",
"Title":                "Trend Micro Console Access"
}
```

5. Create the RFC, specifying the AddTrendMicroLoginRfc file and the AddTrendMicroLoginParams file:

```
aws amscm create-rfc --cli-input-json file://AddTrendMicroLoginRfc.json  --
execution-parameters file://AddTrendMicroLoginParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0wspy4o646g9p.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateEPSDSMReadOnlyUser",
  "Region": "us-east-1",
  "Parameters": {
  "Username": ["eps-dsm-read-only-user"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-CreateEPSDSMReadOnlyUser",
  "Region": "us-east-1",
  "Parameters": {
  "Username": ["eps-dsm-read-only-user"],
  "FullName": ["Alejandro Rosalez"],
  "Description": ["This user is created for eps read only access"]
  }
}
```

Managed Account Subcategory

Change Type Items and Operations in the Managed Account Subcategory

- Automated IAM Provisioning With Read-Write Permissions | Enable (Review Required)
- Automated IAM Provisioning With Read-Write Permissions | Update Custom Deny List (Review Required)
- Developer Mode | Enable (Review Required)
- Direct Change Mode | Enable
- DNS | Migrate To Route 53
- Stack Access Duration | Override (Review Required)

Automated IAM Provisioning With Read-Write Permissions | Enable (Review Required)

Enable Automated IAM provisioning with read-write permissions in the account used to submit this CT. Once enabled, a new role 'AWSManagedServicesIAMProvisionAdminRole' is created in that account. Additionally, you can use three related change types (ct-1n9gfnog5x7fl, ct-1e0xmuy1diafq, ct-17cj84y7632o6) to create, update, or delete IAM roles and policies using Automated IAM provisioning with read-write permissions, which employs an automated review process with a predefined set of rules for IAM and AMS. Before using, we recommend a good familiarity with IAM rules. To confirm that an account has Automated IAM provisioning enabled, look for the IAM role 'AWSManagedServicesIAMProvisionAdminRole' in the IAM console for that account.

Full classification: Management | Managed account | Automated IAM provisioning with read-write permissions | Enable (review required)

Change Type Details

Change type ID	ct-1706xvvk6j9hf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create IAM entity or policy

Creating IAM entity or policy with the console

Create Entity or Policy (read-write permissions)

ID	Execution mode	Version
ct-1n9gfnog5x7fl	Automated	1.0 (only version)

Classification

Deployment -> Advanced stack components -> Identity and Access Management (IAM) -> Create entity or policy (read-write permissions)

Description

Create Identity and Access Management (IAM) role or policy with read-write permissions. You must have enabled this feature with change type ct-1706xvvk6j9hf before submitting this request. Automated IAM provisioning with read-write permissions runs over 200 validations to help ensure successful outcomes.

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating IAM entity or policy with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1n9gfnog5x7fl" --change-type-
version "1.0" --title "Create role or policy" --execution-parameters
 '{"DocumentName":"AWSManagedServices-HandleAutomatedIAMProvisioningCreate-
Admin","Region":"us-east-1","Parameters":{"ValidateOnly":"No"},"RoleDetails":
{"Roles":[{"RoleName":"RoleTest01","Description":"This is a test
 role","AssumeRolePolicyDocument":"{"Version":"2012-10-17","Statement":[{"Effect":"Allow","Prin
{"AWS":"arn:aws:iam::123456789012:root"},"Action":"sts:AssumeRole"}]}","ManagedPolicyArns":
["arn:aws:iam::123456789012:policy/policy01","arn:aws:iam::123456789012:policy/
policy02"],"Path":"/","MaxSessionDuration":"7200","PermissionsBoundary":"arn:aws:iam::12345678
permission_boundary01","InstanceProfile":"No"}]},"ManagedPolicyDetails":
{"Policies":[{"ManagedPolicyName":"TestPolicy01","Description":"This is customer
 policy","Path":"/test/","PolicyDocument":"{"Version":"2012-10-17","Statement":
[{"Sid":"AllQueueActions","Effect":"Allow","Action":"sqs:ListQueues","Resource":"*","Condition"
{"ForAllValues:StringEquals":{"aws:tagKeys":["temporary"]}}}]}"}]}}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; example names it CreateIamResourceParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1n9gfnog5x7fl"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateIamResourceParams.json
```

2. Modify and save the CreateIamResourceParams file; example creates an IAM Role with policy documents pasted inline.

```
{
  "DocumentName": "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin",
  "Region": "us-east-1",
  "Parameters": {
    "ValidateOnly": "No"
  },
  "RoleDetails": {
    "Roles": [
      {
        "RoleName": "RoleTest01",
        "Description": "This is a test role",
        "AssumeRolePolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":
[{\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:root\"},
\"Action\":\"sts:AssumeRole\"}]}",
        "ManagedPolicyArns": [
          "arn:aws:iam::123456789012:policy/policy01",
          "arn:aws:iam::123456789012:policy/policy02"
        ],
        "Path": "/",
        "MaxSessionDuration": "7200",
        "PermissionsBoundary": "arn:aws:iam::123456789012:policy/
permission_boundary01",
        "InstanceProfile": "No"
      }
    ]
  },
  "ManagedPolicyDetails": {
    "Policies": [
      {
        "ManagedPolicyName": "TestPolicy01",
        "Description": "This is customer policy",
        "Path": "/test/",
```

```
        "PolicyDocument": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":
\"AllQueueActions\",\"Effect\":\"Allow\",\"Action\":\"sqs:ListQueues\",\"Resource
\":\"*\",\"Condition\":{\"ForAllValues:StringEquals\":{\"aws:tagKeys\":[\"temporary
\"]}}}]}"
      }
    ]
  }
}
```

3. Output the RFC template JSON file to a file named CreateIamResourceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CreateIamResourceRfc.json
```

4. Modify and save the CreateIamResourceRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1n9gfnog5x7fl",
  "Title": "Create entity or policy (read-write permissions)"
}
```

5. Create the RFC, specifying the CreateIamResourceRfc file and the CreateIamResourceParams file:

```
aws amscm create-rfc --cli-input-json file://CreateIamResourceRfc.json  --
execution-parameters file://CreateIamResourceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- After an IAM role is provisioned in your account, depending on the role and the policy document you attach to the role, you may need to onboard the role in your federation solution.
- For information about AWS Identity and Access Management, see AWS Identity and Access Management (IAM) and for policy information, see Managed policies and inline policies. For information about AMS permissions, see Deploying IAM resources.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1706xvvk6j9hf.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "SAMLIdentityProviderArns": ["arn:aws:iam::123456789012:saml-provider/customer-saml"],
  "IamEntityArns": ["arn:aws:iam::123456789012:role/test-role-one",
 "arn:aws:iam::123456789012:role/test-role-two"],
  "CustomerCustomDenyActionsList1": "ec2:Create*,ec2:Delete*,sso-admin:*,resource-explorer-2:*",
  "Priority": "High"
}
```

Automated IAM Provisioning With Read-Write Permissions | Update Custom Deny List (Review Required)

Update the list of customer-defined denied actions for Automated IAM Provisioning. Make sure to provide the complete list of deny actions, including previously provisioned actions. The provided list replaces the previous list.

Full classification: Management | Managed account | Automated IAM provisioning with read-write permissions | Update custom deny list (review required)

Change Type Details

Change type ID	ct-2r9xvd3sdsic0
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required

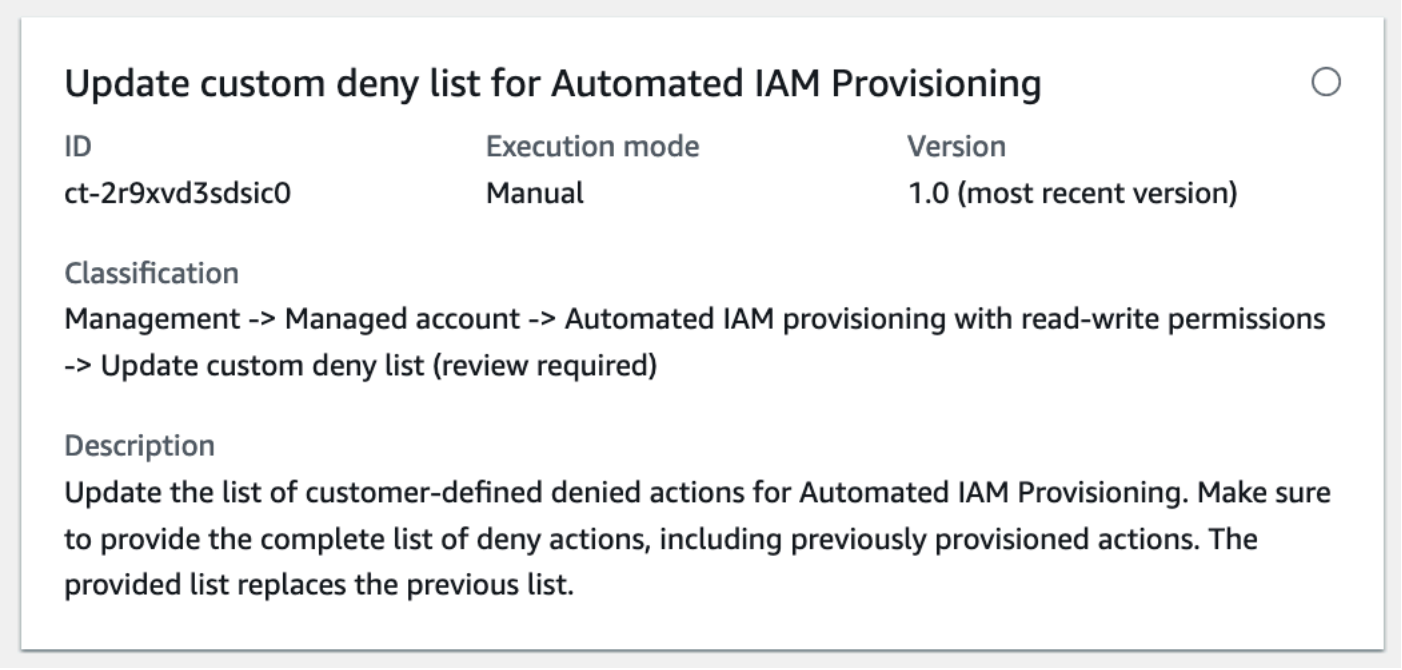
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update custom deny list for AMS Automated IAM Provisioning

Update custom deny list with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Update custom deny list with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2r9xvd3sdsic0" --change-type-version "1.0" --
title "Update custom deny list for Automated IAM Provisioning" --execution-parameters
 "{\"CustomerCustomDenyActionsList1\":\"ec2:RunInstances,s3:PutBucket,sagemaker:*\",
\"Priority\":\"High\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CustomerCustomDenyActionsList.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2r9xvd3sdsic0"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CustomerCustomDenyActionsList.json
   ```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

   ```
   {
       "DocumentName": "AWSManagedServices-CustomerCustomDenyActionsList",
       "Region": "us-east-1",
       "Parameters": {
     "CustomerCustomDenyActionsList1": "ec2:RunInstances,s3:PutBucket,sagemaker:*",
     "Priority": "High"
   }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it CustomerCustomDenyActionsListRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton >
 CustomerCustomDenyActionsListRfc.json
```

4. Modify and save the CustomerCustomDenyActionsListRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2r9xvd3sdsic0",
  "Title": "Update custom deny list for Automated IAM Provisioning"
}
```

5. Create the RFC, specifying the CreateAcmPublicRfc file and the CreateAcmPublicParams file:

```
aws amscm create-rfc --cli-input-json file://CustomerCustomDenyActionsListRfc.json
 --execution-parameters file://CustomerCustomDenyActionsListParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2r9xvd3sdsic0.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "CustomerCustomDenyActionsList1": "ec2:Create*,ec2:Delete*,sso-admin:*,resource-
explorer-2:*",
  "Priority": "High"
```

```
}
```

Developer Mode | Enable (Review Required)

Enable Developer Mode (Dev Mode). Dev mode provides you with elevated permissions, in AMS Plus accounts, to provision and update AWS resources outside of the AMS change management process. Dev mode does this by leveraging native AWS API calls within the AMS Virtual Private Cloud (VPC), enabling you to design and implement infrastructure and applications in your managed environment. When using an account that has Dev mode enabled, continuity management, patch management, and change management are provided for resources provisioned through the AMS change management process or by using an AMS Amazon Machine Image (AMI). However, these AMS management features are not offered for resources provisioned through native AWS APIs. Rather, you are responsible for monitoring infrastructure resources that are provisioned outside of the AMS change management process. Dev mode is limited to accounts with non-production workloads. With elevated permissions, you have an increased responsibility to ensure adherence to internal controls.

Full classification: Management | Managed account | Developer mode | Enable (review required)

Change Type Details

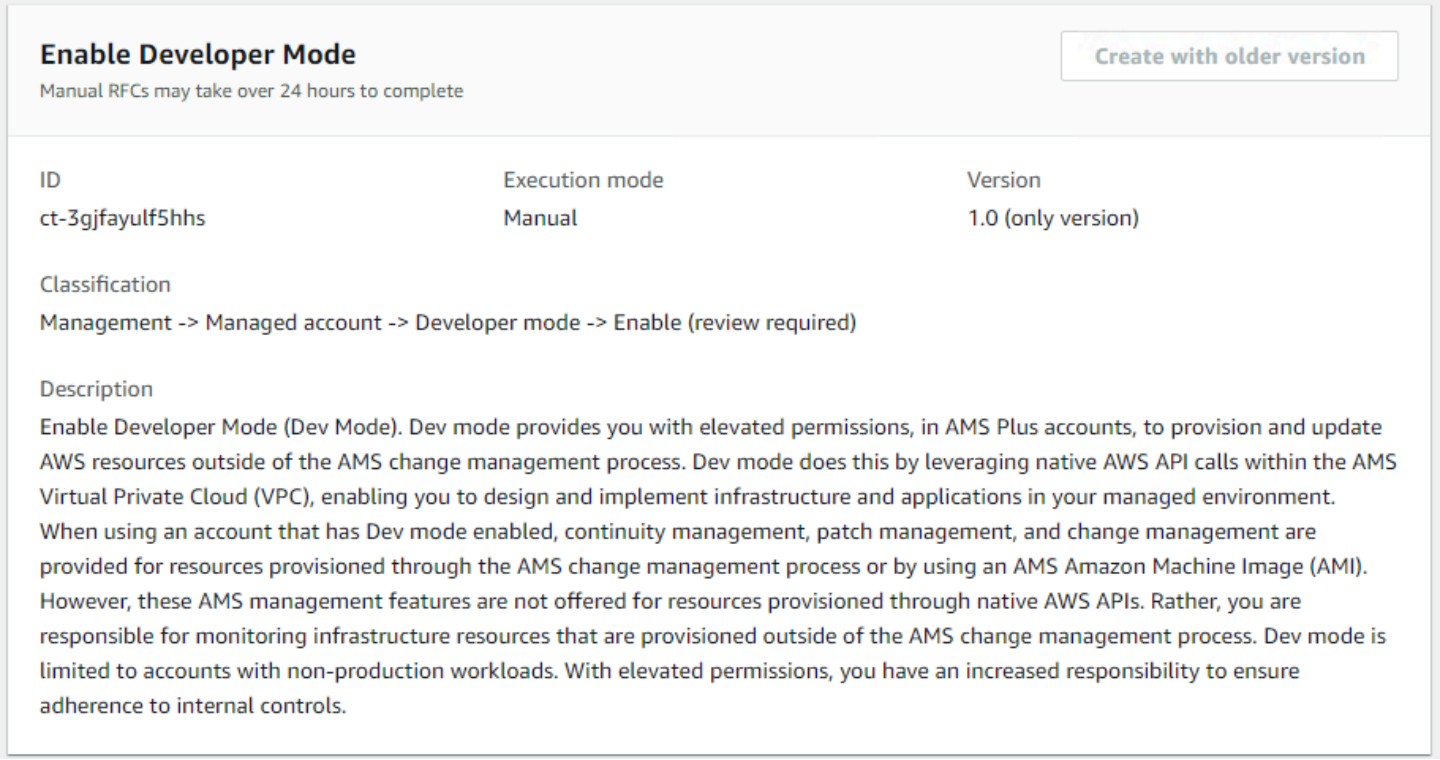
Change type ID	ct-3gjfayulf5hhs
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Enable Developer mode (review required)

Enabling Developer mode (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling Developer mode (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Application account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3gjfayulf5hhs" --change-type-version "1.0" --
title "RFC Title" --execution-parameters "{\"Enable\":\"Yes\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it EnableDevModeParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3gjfayulf5hhs" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > EnableDevModeParams.json
   ```

2. Modify and save the EnableDevModeParams file. For example, you can replace the contents with something like this:

   ```
   {"Enable": "Yes"}
   ```

3. Output the RFC template JSON file to a file; this example names it EnableDevModeRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableDevModeRfc.json
   ```

4. Modify and save the EnableDevModeRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-3gjfayulf5hhs",
   "Title":                "Enable-Dev-Mode-RFC"
   }
   ```

5. Create the RFC, specifying the EnableDevModeRfc file and the EnableDevModeParams file:

```
aws amscm create-rfc --cli-input-json file://EnableDevModeRfc.json  --execution-
parameters file://EnableDevModeParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

> **Note**
>
> When using "review required" CTs, AMS recommends that you use the ASAP **Scheduling** option (choose **ASAP** in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run. If you schedule these RFCs, be sure to allow at least 24 hours. If approval does not happen before the scheduled start time, the RFC is rejected automatically.

- Resources that you create using developer mode can be managed by AMS only if they are created using AMS change management processes.
- For more information about Developer mode, see AMS Developer Mode.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3gjfayulf5hhs.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Enable": "Yes",
  "Priority": "Medium"
}
```

Direct Change Mode | Enable

Enable Direct Change mode (DCM). DCM grants native AWS access to provision and update AWS resources. The resources and changes to them are fully supported by AMS, including monitoring, patch, backup, and incident response management.

Full classification: Management | Managed account | Direct Change mode | Enable

Change Type Details

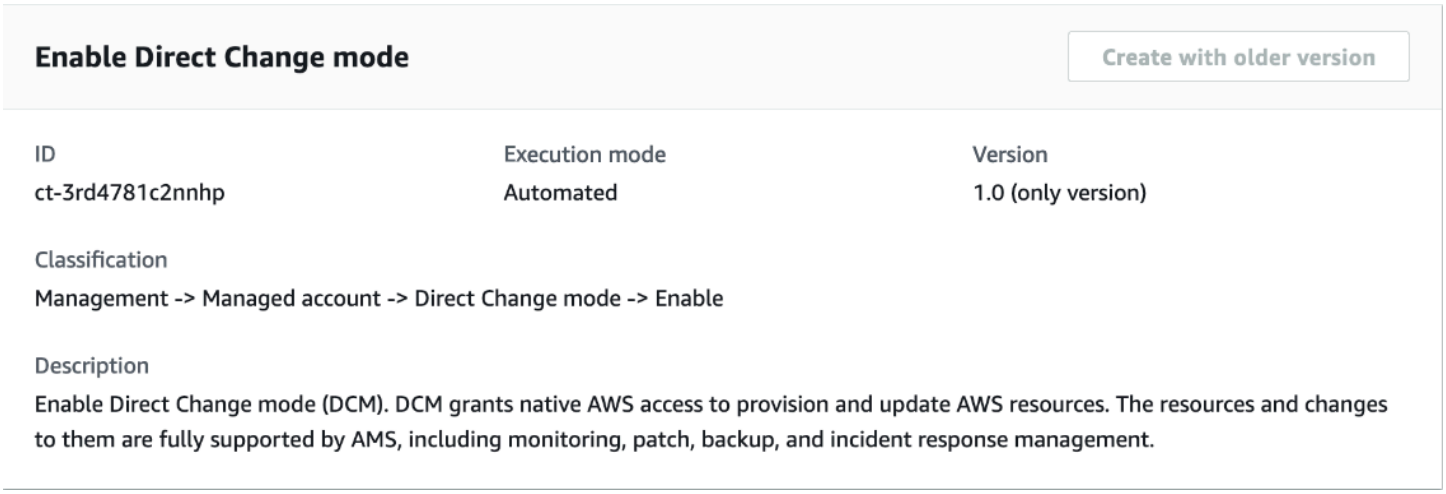
Change type ID	ct-3rd4781c2nnhp
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable Direct Change mode

Enabling Direct Change mode with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling Direct Change mode with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
```

```
--change-type-id "ct-3rd4781c2nnhp" \
--change-type-version "1.0" \
--title "Enable Direct Change Mode" \
--execution-parameters "{\"samlIdentityProviderArns\":\"arn:aws:iam::123456789123:saml-provider/valid-name\",\"iamEntityArns\":\"arn:aws:iam::123456789123:role/valid-role-name1\",\"awsServicePrincipals\":\"ec2.amazonaws.com\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it EnableDCMParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3rd4781c2nnhp" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > EnableDCMParams.json
   ```

2. Modify and save the EnableDCMParams file. For example, you can replace the contents with something like this:

   ```
   {
     "samlIdentityProviderArns": "arn:aws:iam::123456789123:saml-provider/valid-name",
     "iamEntityArns": "arn:aws:iam::123456789123:role/valid-role-name,
     "awsServicePrincipals": "ec2.amazonaws.com"
   }
   ```

3. Output the RFC template JSON file to a file named EnableDCMRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableDCMRfc.json
   ```

4. Modify and save the EnableDCMRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-3rd4781c2nnhp",
   "Title":                "Enable-DCM-RFC"
   }
   ```

5. Create the RFC, specifying the EnableDCMRfc file and the EnableDCMParams file:

   ```
   aws amscm create-rfc --cli-input-json file://EnableDCMRfc.json  --execution-
   parameters file://EnableDCMParams.json
   ```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Direct Change mode, see Direct Change Mode.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3rd4781c2nnhp.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "SamlIdentityProviderArns": [
    "SAML1",
    "SAML2"
  ],
  "AwsServicePrincipals": [
    "Service1",
    "Service2"
  ],
  "IamEntityArns": [
    "role1",
    "role2"
  ]
}
```

DNS | Migrate To Route 53

Change the DNS resolution in your Amazon VPC by enabling Route 53 as the default DNS resolver for your SALZ account. This transition from Microsoft AD to Route 53 Resolver involves redirecting DNS traffic within your VPC through strategically implemented Route 53 Resolver Endpoints and Conditional Forwarders. These forwarders act as rules to intelligently route DNS queries, ensuring

seamless resolution for various destinations. It's essential to plan the migration during a scheduled maintenance window to minimize potential disruptions caused by DNS changes.

Full classification: Management | Managed account | DNS | Migrate to Route 53

Change Type Details

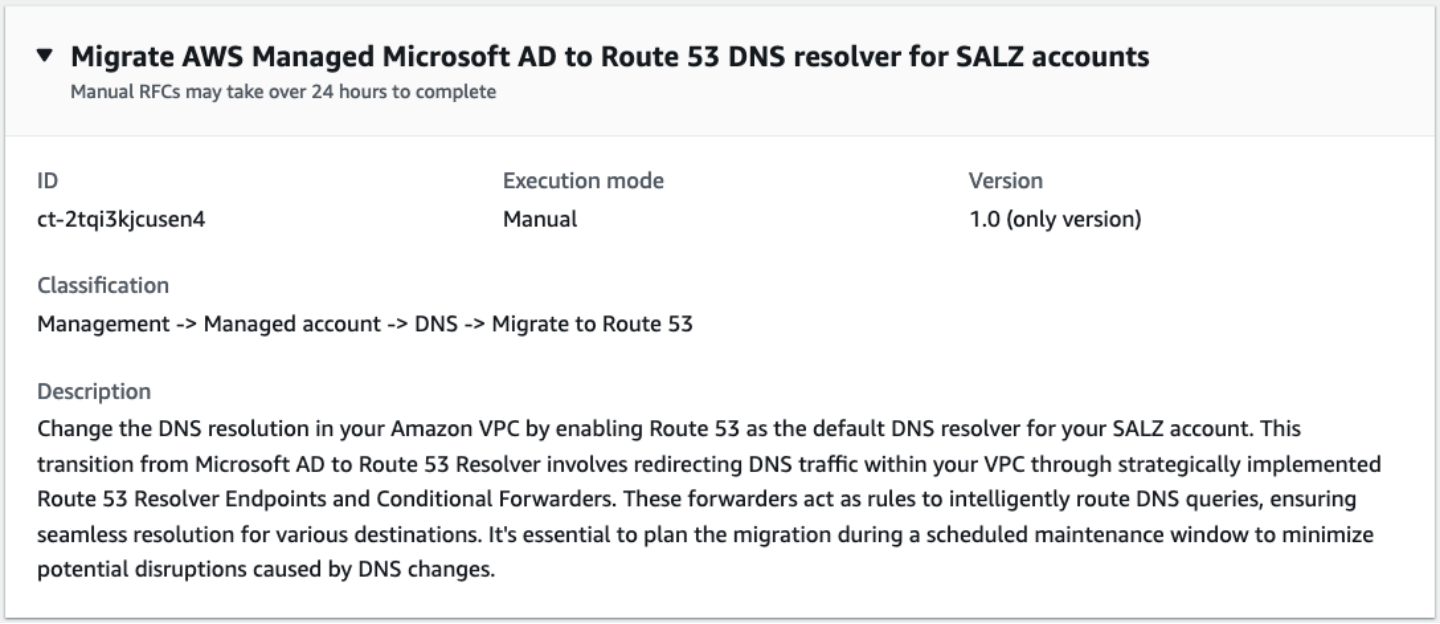
Change type ID	ct-2tqi3kjcusen4
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Migrate AWS Managed Microsoft AD to Route 53 DNS resolver for SALZ accounts

Migrate AWS Managed Microsoft AD to Route 53 with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Migrate AWS Managed Microsoft AD to Route 53 DNS resolver with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

Required parameters only:

```
aws amscm create-rfc --change-type-id "ct-2tqi3kjcusen4" --change-type-version "1.0" --
title "Migrate AWS managed Microsoft AD to Route 53 DNS resolver for SALZ accounts" --
execution-parameters "{}"
```

All required and optional parameters:

```
aws amscm create-rfc --change-type-id "ct-2tqi3kjcusen4" --change-type-version "1.0" --
title "Migrate AWS managed Microsoft AD to Route 53 DNS resolver for SALZ accounts" --
execution-parameters "{\"Priority\":\"Medium\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named CreateMigrateToRoute53RequiredParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-2tqi3kjcusen4"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 CreateMigrateToRoute53RequiredParams.json
```

2. Modify and save the execution parameters JSON file. For example, you can replace the contents with something like this:

```
{
  "Priority": "Medium"
}
```

3. Output the RFC template to a file in your current folder; this example names it CreateMigrateToRoute53RequiredRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton >
 CreateMigrateToRoute53RequiredRfc.json
```

4. Modify and save the CreateMigrateToRoute53RequiredRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId":          "ct-2tqi3kjcusen4",
"ChangeTypeVersion":     "1.0",
"Title":                 "Migrate AWS managed Microsoft AD to Route 53 DNS resolver
 for SALZ accounts"
}
```

5. Create the RFC, specifying the CreateMigrateToRoute53RequiredRfc file and the CreateMigrateToRoute53RequiredParams file:

```
aws amscm create-rfc --cli-input-json file://CreateMigrateToRoute53RequiredRfc.json
  --execution-parameters file://CreateMigrateToRoute53RequiredParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2tqi3kjcusen4.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

Stack Access Duration | Override (Review Required)

Use to override maximum stack access time for all stacks in this account for single landing zone (SALZ) and for all stacks of the member accounts of an organization for multi-landing zone (MALZ). For multi-landing zone (MALZ), please raise a request for change (RFC) from shared-services account with this change type (CT) ID. Access can be overridden from a minimum of 1 hour to a maximum of 120 hours, default stack access is granted for 12 hours.

Full classification: Management | Managed account | Stack access duration | Override (review required)

Change Type Details

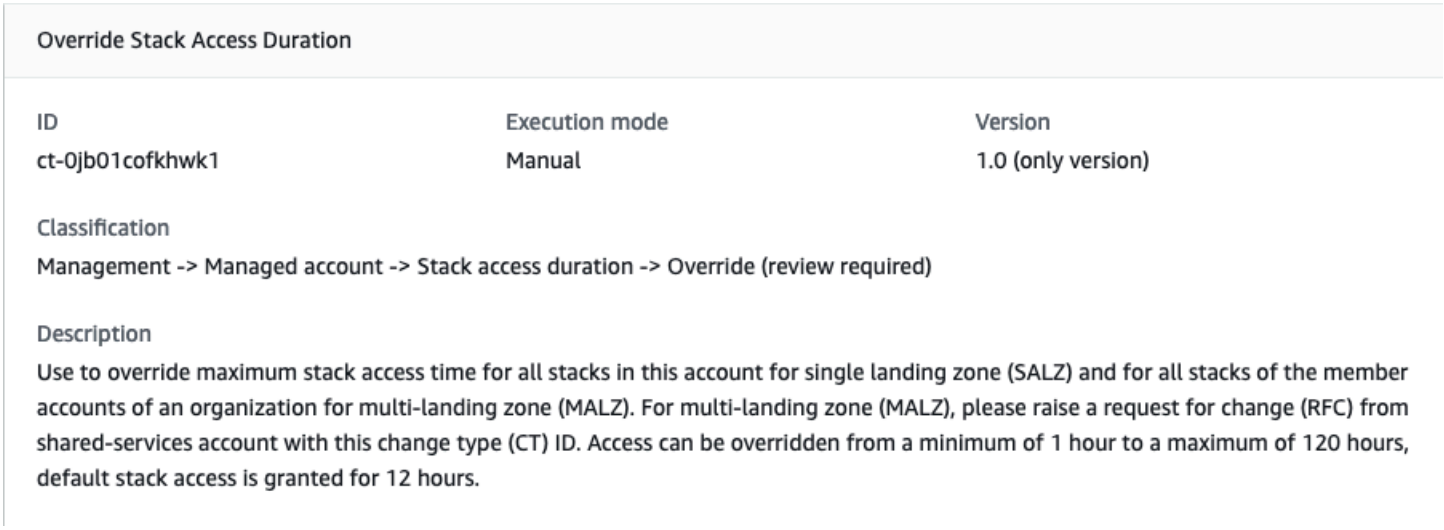
Change type ID	ct-0jb01cofkhwk1
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Override Stack Access Duration (Review required)

Override stack access duration with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Override stack access duration with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --title="Override Stack Access Duration" --description="Override
 Stack Access Duration" --ct-id="ct-0jb01cofkhwk1" --ct-version="1.0" --input-
params="{\"TimeRequestedInHours\": 15,\"Priority\":\"High\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file. This example names it OverrideStackAccessDurationParameters.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0jb01cofkhwk1"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    OverrideStackAccessDurationParameters.json
   ```

2. Modify and save the OverrideStackAccessDurationParameters.json file. For example, you can replace the contents with something like this:

   ```
   {
     "TimeRequestedInHours": 15,
     "Priority": "High"
   }
   ```

3. Output the RFC template JSON file to a file named OverrideStackAccessDuration.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > OverrideStackAccessDuration.json
   ```

4. Modify and save the OverrideStackAccessDuration.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-0jb01cofkhwk1",
     "Title": "Override Stack Access Duration"
   }
   ```

5. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://OverrideStackAccessDuration.json --
execution-parameters file://OverrideStackAccessDurationParameters.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0jb01cofkhwk1.

Example: Required Parameters

```
{
    "TimeRequestedInHours": 120
}
```

Example: All Parameters

```
{
    "TimeRequestedInHours": 15,
    "Priority": "High"
}
```

Managed Firewall Subcategory

Change Type Items and Operations in the Managed Firewall Subcategory

- Outbound (Palo Alto) | Add URLs
- Outbound (Palo Alto) | Delete Allow List
- Outbound (Palo Alto) | Delete Security Policy
- Outbound (Palo Alto) | Remove URLs
- Outbound (Palo Alto) | Update Security Policy

Outbound (Palo Alto) | Add URLs

Add allow list URLs for AMS managed Palo Alto firewall - Outbound.

Full classification: Management | Managed Firewall | Outbound (Palo Alto) | Add URLs

Change Type Details

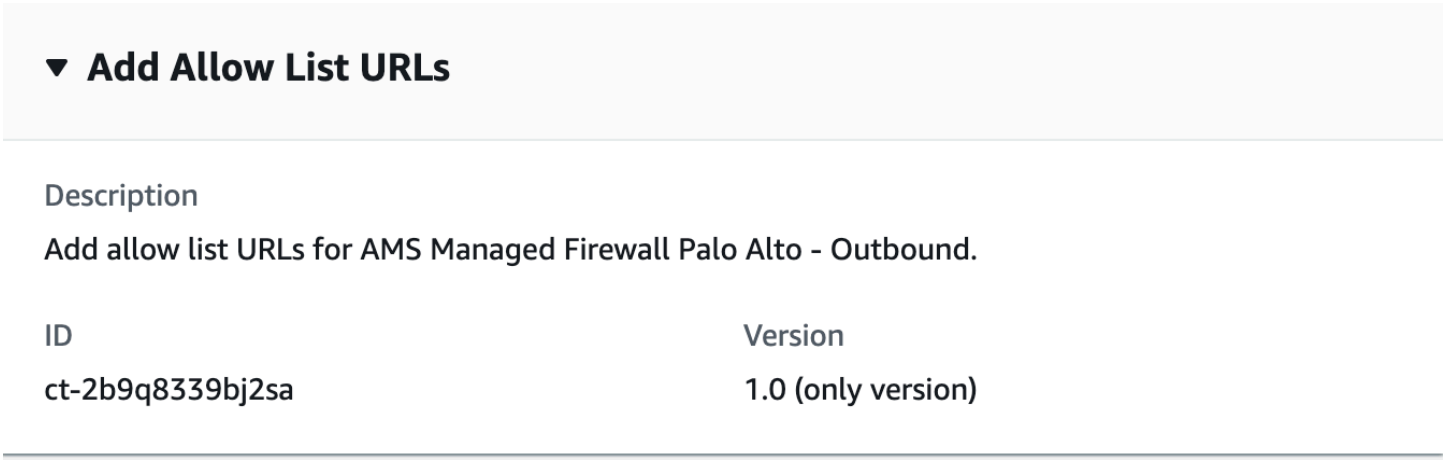
Change type ID	ct-2b9q8339bj2sa
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Add URLs to managed Palo Alto outbound firewall

Adding outbound URLs to your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Adding outbound URLs to your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2b9q8339bj2sa" --change-type-version "2.0" --
title "Add URLs to Allow List" --execution-parameters "{ \"RequestType\": \"AddURLs\",
 \"Parameters\": { \"URLs\": [ \"amazon.com/\", \"*.website.com\\" ], "AllowListName":
 "CustomAllowList" } } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AddPaUrlsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2b9q8339bj2sa" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > AddPaUrlsParams.json
   ```

2. Modify and save the AddPaUrlsParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "AddURLs",
     "Parameters": {
         "URLs": [
             "amazon.com/",
   ```

```
            "*.website.com/"
        ],
        "AllowListName": "CustomAllowList"
    }
}
```

3. Output the RFC template JSON file to a file named AddPaUrlsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > AddPaUrlsRfc.json
```

4. Modify and save the AddPaUrlsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2b9q8339bj2sa",
"Title":                "Add-Urls-RFC"
}
```

5. Create the RFC, specifying the AddPaUrls Rfc file and the AddPaUrlsParams file:

```
aws amscm create-rfc --cli-input-json file://AddPaUrlsRfc.json  --execution-
parameters file://AddPaUrlsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change has a new version and a new schema.

> **Note**
>
> If you are a Beta customer for AMS Palo Alto managed firewall, do not use this change type, it does not work with Beta accounts. Use the Management | Other | Other | Update (ct-0xdawir96cy7k) instead.

To learn more about Palo Alto managed firewall in AMS, seeManaged Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2b9q8339bj2sa.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "RequestType": "AddURLs",
  "Parameters": {
    "URLs": ["amazon.com/","*.amazon.com/","www.*.com/","amazon.co1m/","amazon.c-
m/","ama-zon.com/","long.sub.domain.amazon.com/","long.sub.*.amazon.com/"],
    "AllowListName": "test_file"
  }
}
```

Outbound (Palo Alto) | Delete Allow List

Delete an allow list file for AMS managed Palo Alto firewall - Outbound.

Full classification: Management | Managed Firewall | Outbound (Palo Alto) | Delete allow list

Change Type Details

Change type ID	ct-2fzh1wckpl7f5
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete allow list from managed Palo Alto outbound firewall

Deleting an allow list from your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an allow list from your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2fzh1wckpl7f5" --change-type-version "1.0" --
title "Delete Allow List" --execution-parameters "{ \"RequestType\": \"DeleteAllowList
\", \"Parameters\": { "AllowListName": "CustomAllowList" } } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteAllowListParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-2mf36chtp1ejh"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeleteAllowListParams.json
```

2. Modify and save the DeleteAllowListParams file. For example, you can replace the contents with something like this:

```
{
  "RequestType": "DeleteAllowList",
  "Parameters": {
      "AllowListName": "CustomAllowList"
  }
}
```

3. Output the RFC template JSON file to a file named DeleteAllowListRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteAllowListRfc.json
```

4. Modify and save the DeleteAllowListRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2fzh1wckpl7f5",
"Title":                "Delete-Allow-List-RFC"
}
```

5. Create the RFC, specifying the DeleteAllowList Rfc file and the DeleteAllowListParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteAllowListRfc.json  --execution-
parameters file://DeleteAllowListParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If you are a Beta customer for AMS Palo Alto managed firewall, do not use this change type, it does not work with Beta accounts. Use the Management | Other | Other | Update (ct-0xdawir96cy7k) instead.

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2fzh1wckpl7f5.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Parameters": {
    "AllowListName": "test_file"
  },
  "RequestType": "DeleteAllowList"
}
```

Outbound (Palo Alto) | Delete Security Policy

Delete a security policy for AMS managed Palo Alto firewall - Outbound.

Full classification: Management | Managed Firewall | Outbound (Palo Alto) | Delete security policy

Change Type Details

Change type ID	ct-1taxucdyi84iy

Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete security policy from managed Palo Alto outbound firewall

Deleting a security policy from your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting a security policy from your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1taxucdyi84iy" --change-type-version
 "1.0" --title "Delete Security Policy" --execution-parameters "{ \"RequestType\":
 \"DeleteSecurityPolicy\", \"Parameters\": { "SecurityPolicyName": "custom-sec-
name" } } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteSecurityPolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1taxucdyi84iy"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteSecurityPolicyParams.json
   ```

2. Modify and save the DeleteSecurityPolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "DeleteSecurityPolicy",
     "Parameters": {
         "SecurityPolicyName": "custom-sec-name"
     }
   }
   ```

3. Output the RFC template JSON file to a file named DeleteSecurityPolicyRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteSecurityPolicyRfc.json
   ```

4. Modify and save the DeleteSecurityPolicyRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-1taxucdyi84iy",
   "Title":                "Delete-Security-Policy-RFC"
   ```

```
}
```

5. Create the RFC, specifying the DeleteSecurityPolicy Rfc file and the DeleteSecurityPolicyParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteSecurityPolicyRfc.json  --
execution-parameters file://DeleteSecurityPolicyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1taxucdyi84iy.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Parameters": {
    "SecurityPolicyName": "custom-sec-pol"
  },
  "RequestType": "DeleteSecurityPolicy"
}
```

Outbound (Palo Alto) | Remove URLs

Remove URLs from an allow list file for AMS managed Palo Alto firewall - Outbound.

Full classification: Management | Managed Firewall | Outbound (Palo Alto) | Remove URLs

Change Type Details

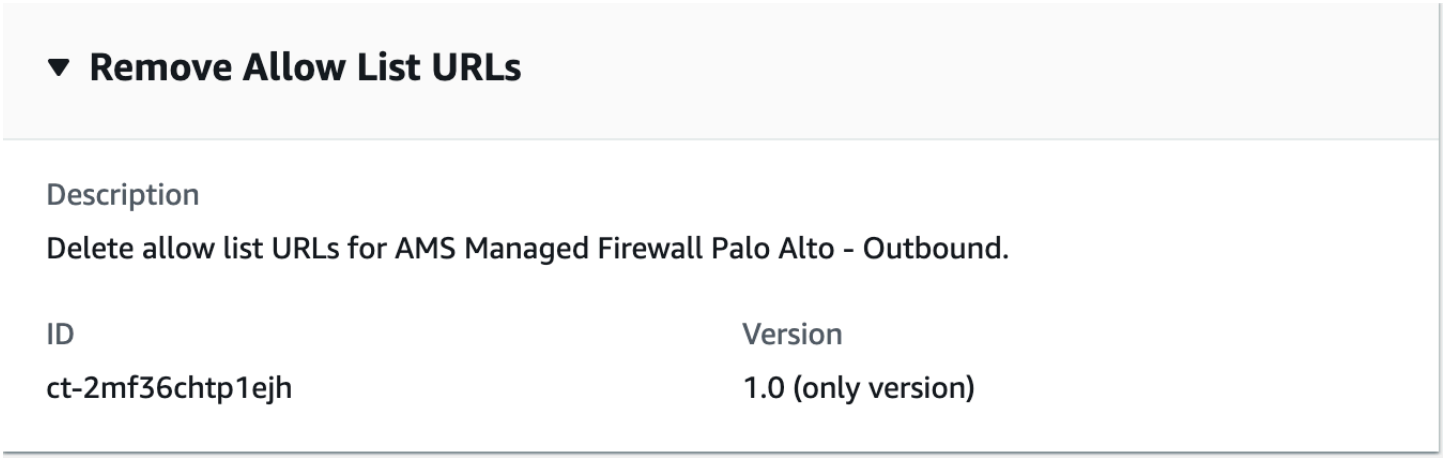
Change type ID	ct-2mf36chtp1ejh
Current version	2.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove URLS from managed Palo Alto outbound firewall

Removing outbound URLs from your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Removing outbound URLs from your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2mf36chtp1ejh" --change-type-version "2.0"
 --title "Remove URLs from Allow List" --execution-parameters "{ \"RequestType\":
 \"RemoveURLs\", \"Parameters\": { \"URLs\": [ \"amazon.com/\", \"*.website.com/\" ],
 "AllowListName": "CustomAllowList" } } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemovePaUrlsParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2mf36chtp1ejh" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RemovePaUrlsParams.json
   ```

2. Modify and save the RemovePaUrlsParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "RemoveURLs",
     "Parameters": {
         "URLs": [
   ```

```
            "amazon.com/",
            "*.website.com/"
        ],
        "AllowListName": "CustomAllowList"
    }
}
```

3. Output the RFC template JSON file to a file named RemovePaUrlsRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RemovePaUrlsRfc.json
```

4. Modify and save the RemovePaUrlsRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2mf36chtp1ejh",
"Title":                "Remove-Urls-RFC"
}
```

5. Create the RFC, specifying the RemovePaUrls Rfc file and the RemovePaUrlsParams file:

```
aws amscm create-rfc --cli-input-json file://RemovePaUrlsRfc.json  --execution-
parameters file://RemovePaUrlsParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change has a new version and a new schema.

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2mf36chtp1ejh.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Parameters": {
    "URLs": ["amazon.com/","*.amazon.com/","www.*.com/","amazon.co1m/","amazon.c-
m/","ama-zon.com/","long.sub.domain.amazon.com/","long.sub.*.amazon.com/"],
    "AllowListName": "test_file"
  },
  "RequestType": "RemoveURLs"
}
```

Outbound (Palo Alto) | Update Security Policy

Update a security policy for AMS managed Palo Alto firewall - Outbound.

Full classification: Management | Managed Firewall | Outbound (Palo Alto) | Update security policy

Change Type Details

Change type ID	ct-0mss4i7neuj7f
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update security policy for managed Palo Alto outbound firewall

Updating a security policy for your managed Palo Alto firewall with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a security policy for your managed Palo Alto firewall with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC

parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *ID* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *ID* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0mss4i7neuj7f" --change-type-version
 "1.0" --title "Update Security Policy" --execution-parameters "{ \"RequestType\":
 \"UpdateSecurityPolicy\", \"Parameters\": { "SecurityPolicyName": "custom-sec-name",
 "SourceAddressesToAdd": ["3.0.0.0"], "DestinationAddressesToAdd": ["4.0.0.0"],
 "AllowListsToAdd": [], "ServicePortsToAdd": { "TCPPortsToAdd": [30] "UDPPortsToAdd":
 [40] }, "SourceAddressesToRemove": [], "DestinationAddressesToRemove": [],
 "AllowListsToRemove": [], "ServicePortsToRemove": { "TCPPortsToRemove": [],
 "UDPPortsToRemove": [] }, "ActionType": "Allow", "EnablePolicy": aws amscm
 create-rfc --change-type-id "ct-0mss4i7neuj7f" --change-type-version "1.0"
 --title "Update Security Policy" --execution-parameters "{ \"RequestType\":
 \"UpdateSecurityPolicy\", \"Parameters\": { "SecurityPolicyName": "custom-
sec-name", "SourceAddressesToAdd": ["3.0.0.0"], "DestinationAddressesToAdd":
```

```
["4.0.0.0"], "AllowListsToAdd": [], "ServicePortsToAdd": { "TCPPortsToAdd":
[30] "UDPPortsToAdd": [40] }, "SourceAddressesToRemove": [SOURCE_ADDRESSES],
"DestinationAddressesToRemove": [DEST_ADDRESSES], "AllowListsToRemove": [],
"ServicePortsToRemove": { "TCPPortsToRemove": [TCP_PORT], "UDPPortsToRemove":
[UDP_PORT] }, "ActionType": "Allow", "EnablePolicy": true } " } "
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateSecurityPolicyParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0mss4i7neuj7f"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateSecurityPolicyParams.json
   ```

2. Modify and save the UpdateSecurityPolicyParams file. For example, you can replace the contents with something like this:

   ```
   {
     "RequestType": "UpdateSecurityPolicy",
     "Parameters": {
         "SecurityPolicyName": "custom-sec-name",
         "SourceAddressesToAdd": ["3.0.0.0"],
         "DestinationAddressesToAdd": ["4.0.0.0"],
         "AllowListsToAdd": [],
         "ServicePortsToAdd": {
               "TCPPortsToAdd": [30],
               "UDPPortsToAdd": [40]
         },
         "SourceAddressesToRemove": [],
         "DestinationAddressesToRemove": [],
         "AllowListsToRemove": [],
         "ServicePortsToRemove": {
               "TCPPortsToRemove": [],
               "UDPPortsToRemove": []
         },
         "ActionType": "Allow",
         "EnablePolicy": true
     }
   }
   ```

3. Output the RFC template JSON file to a file named UpdateSecurityPolicyRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateSecurityPolicyRfc.json
```

4. Modify and save the UpdateSecurityPolicyRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0mss4i7neuj7f",
"Title":                "Update-Security-Policy-RFC"
}
```

5. Create the RFC, specifying the UpdateSecurityPolicy Rfc file and the UpdateSecurityPolicyParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateSecurityPolicyRfc.json  --
execution-parameters file://UpdateSecurityPolicyParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Palo Alto managed firewall in AMS, see Managed Palo Alto egress firewall.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0mss4i7neuj7f.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "Parameters": {
      "SecurityPolicyName": "custom-sec-pol",
      "SourceAddressesToAdd": ["10.0.0.1", "10.50.0.0/16"],
```

```
      "DestinationAddressesToAdd": ["1.1.1.1", "88.0.0.0/8", "amazon.com"],
      "ServicePortsToAdd": {"TCPPortsToAdd": [1000, 1200]},
      "SourceAddressesToRemove": ["10.0.1.1", "10.80.55.0/24"],
      "DestinationAddressesToRemove": ["8.8.8.8", "7.60.33.155/32", "google.com"],
      "ServicePortsToRemove": {"TCPPortsToRemove": [9000, 3333]},
      "ActionType": "Allow",
      "EnablePolicy": false
    },
    "RequestType": "UpdateSecurityPolicy"
}
```

Managed Landing Zone Subcategory

Change Type Items and Operations in the Managed Landing Zone Subcategory

- Application Account | Confirm Offboarding
- Application Account | Delete VPC
- Management Account | Delete StackSets Stack (Review Required)
- Management Account | Enable Developer Mode
- Management Account | Move Account To OU
- Management Account | Offboard Application Account
- Management Account | Update StackSets Stack (Review Required)
- Networking Account | Associate TGW Attachment
- Networking Account | Disable TGW Propagation
- Networking Account | Disassociate TGW Attachment
- Networking Account | Enable TGW Propagation
- Networking Account | Remove TGW Static Route

Application Account | Confirm Offboarding

Confirm offboarding of the specified application account. Run this from the application account that you want offboarded. Once confirmed, run the Execute offboarding CT (ct-0vdiy51oyrhhm) from the associated management account. Note that this offboarding is intended for account closure and cannot be undone

Full classification: Management | Managed landing zone | Application account | Confirm offboarding

Change Type Details

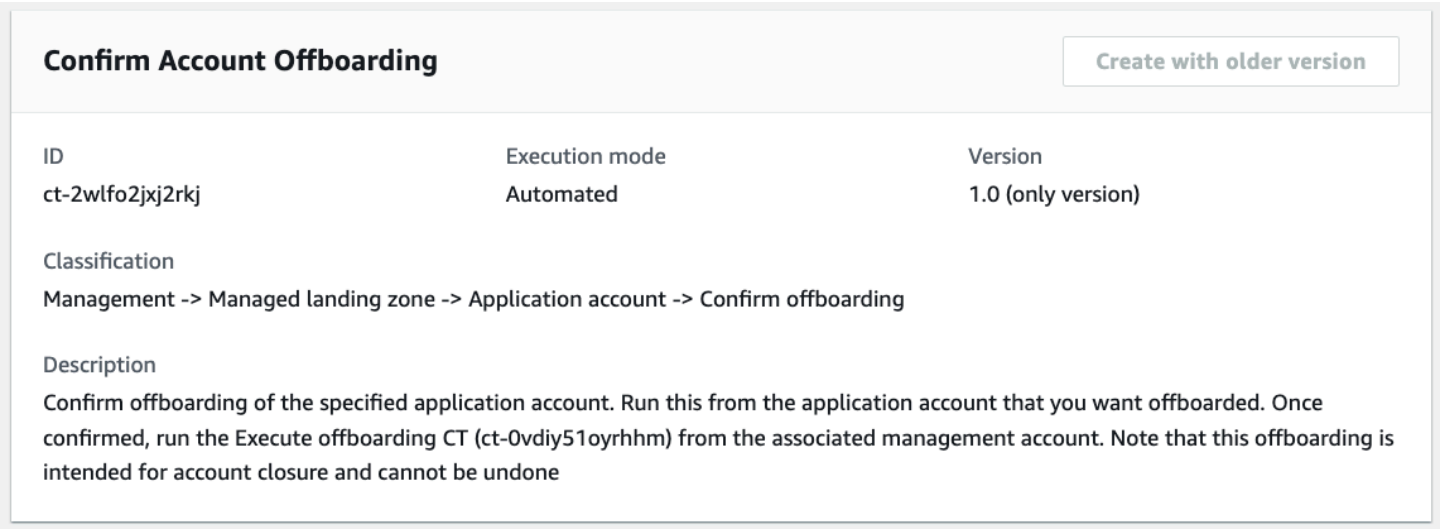
Change type ID	ct-2wlfo2jxj2rkj
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Confirm offboarding

Application account: Confirming offboarding with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Application account: Confirming offboarding with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Application account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2wlfo2jxj2rkj" --change-type-version "1.0" --
title "Confirm Offboarding" --execution-parameters "{\"AccountID\": \"000000000000\",
\"AccountEmail\": \"email@amazon.com\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it ConfirmAppAcctOffBParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2wlfo2jxj2rkj"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    ConfirmAppAcctOffBParams.json
   ```

2. Modify and save the ConfirmAppAcctOffBParams file. For example, you can replace the contents with something like this:

```
{
  "AccountID": "000000000000",
  "AccountEmail": "email@amazon.com",
}
```

3. Output the RFC template JSON file to a file; this example names it ConfirmAppAcctOffBRfc.json:

```
    aws amscm create-rfc --generate-cli-skeleton > ConfirmAppAcctOffBRfc.json
```

4. Modify and save the ConfirmAppAcctOffBRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-2wlfo2jxj2rkj",
  "Title": "Confirm Offboarding"
}
```

5. Create the RFC, specifying the ConfirmAppAcctOffBRfc file and the ConfirmAppAcctOffBParams file:

```
    aws amscm create-rfc --cli-input-json file://ConfirmAppAcctOffBRfc.json  --
execution-parameters file://ConfirmAppAcctOffBParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For application accounts (other than Customer Managed), run this from the Application account that you want offboarded. Once confirmed, run the Offboard application account CT (ct-0vdiy51oyrhhm) from the associated management account. Offboarding is intended for account closure and cannot be undone.
- Do not use this CT for Customer Managed application accounts. Go directly to Offboard application account CT (ct-0vdiy51oyrhhm).

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2wlfo2jxj2rkj.

Example: Required Parameters

```
{
  "RequestType": "OffboardingConfirmation",
  "Parameters": {
    "AccountId": "000000000000",
    "AccountEmail": "example@email.com"
  }
}
```

Example: All Parameters

```
{
  "RequestType": "OffboardingConfirmation",
  "Parameters": {
    "AccountId": "000000000000",
    "AccountEmail": "example@email.com"
  }
}
```

Application Account | Delete VPC

Delete the virtual private cloud (VPC) in a managed landing zone application account.

Full classification: Management | Managed landing zone | Application account | Delete VPC

Change Type Details

Change type ID	ct-2paw0y79kvr3l
Current version	1.0
Expected execution duration	360 minutes
AWS approval	Required

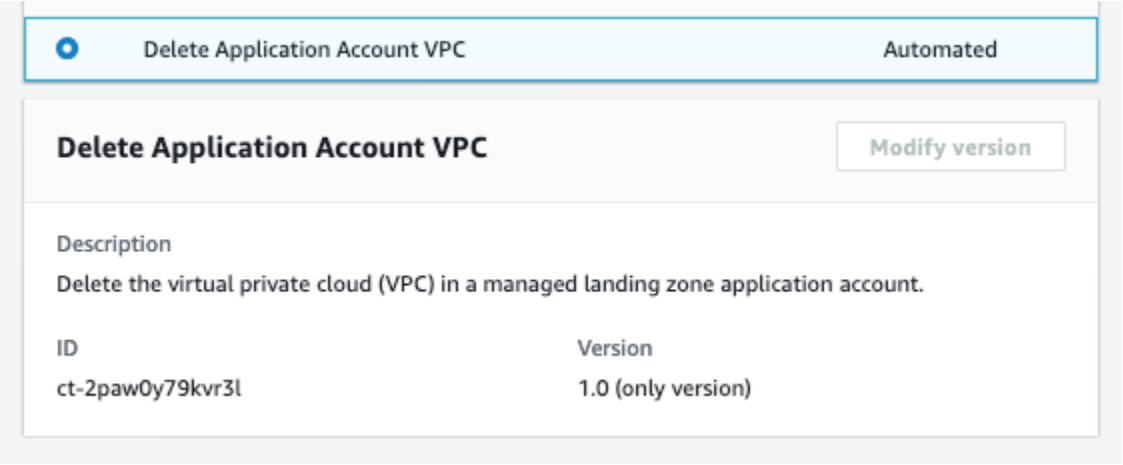
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete VPC

Application account: deleting a VPC with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Application account: deleting a VPC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Application account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
       aws amscm create-rfc --change-type-id "ct-2paw0y79kvr3l" --change-type-version
"1.0" --title "Delete VPC" --execution-parameters "{\"VPCId\":\"VPC_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteAppAcctVpcParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2paw0y79kvr3l"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteAppAcctVpcParams.json
   ```

2. Modify and save the DeleteAppAcctVpcParams file. For example, you can replace the contents with something like this:

   ```
       {
   "VPCId": "VPC_ID"
       }
   ```

3. Output the RFC template JSON file to a file; this example names it DeleteAppAcctVpcRfc.json:

```
    aws amscm create-rfc --generate-cli-skeleton > DeleteAppAcctVpcRfc.json
```

4. Modify and save the DeleteAppAcctVpcRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion" : "1.0",
"ChangeTypeId" : "ct-2paw0y79kvr3l",
"Title" : "App-Acct-Vpc-RFC"
}
```

5. Create the RFC, specifying the DeleteAppAcctVpcRfc file and the DeleteAppAcctVpcParams file:

```
    aws amscm create-rfc --cli-input-json file://DeleteAppAcctVpcRfc.json  --
execution-parameters file://DeleteAppAcctVpcParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about VPCs, see Working with VPCs and subnets.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2paw0y79kvr3l.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "VPCId": "vpc-0078e69aa52274dea"
}
```

Management Account | Delete StackSets Stack (Review Required)

Delete AWS CloudFormation (CFN) StackSets-created stacks and instances.

Full classification: Management | Managed landing zone | Management account | Delete StackSets stack (review required)

Change Type Details

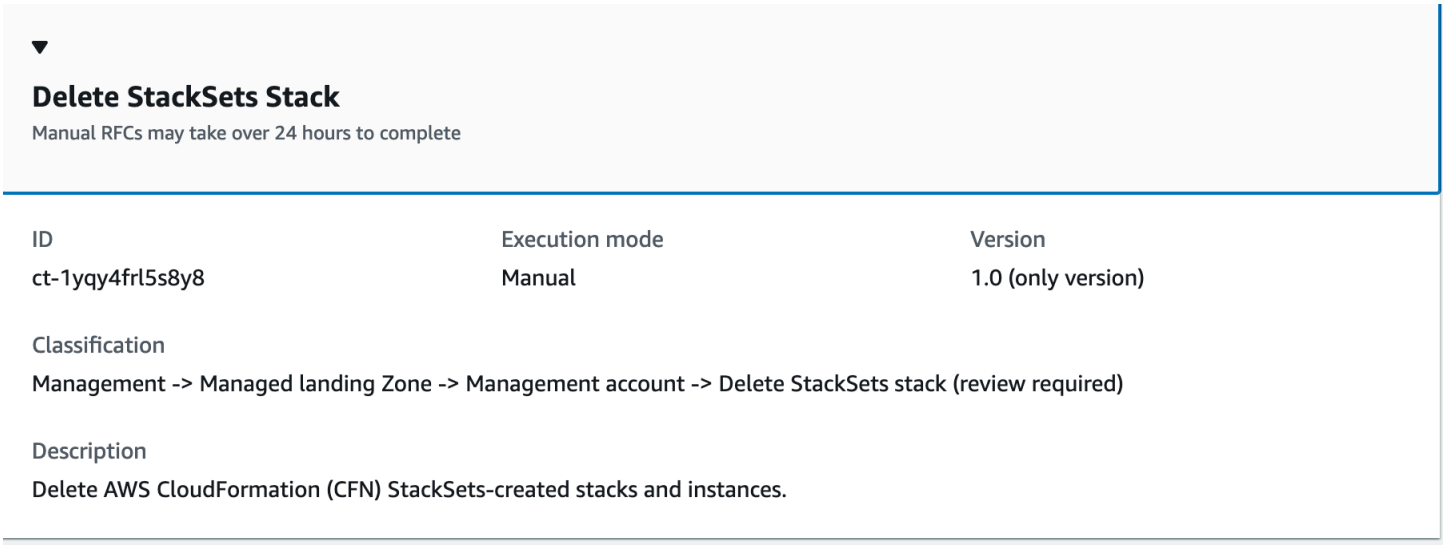
Change type ID	ct-1yqy4frl5s8y8
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete a Stacksets stack

Deleting a Stacksets stack with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting a Stacksets stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id ID` command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1yqy4frl5s8y8" --change-type-version "1.0" --
title "Delete StackSets Stack" --execution-parameters "{\"Name\": \"Stackset name\",
 \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteStacksetsStackParams.json:

```
aws amscm get-change-type-version --change-type-id "ct-1yqy4frl5s8y8"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 DeleteStacksetsStackParams.json
```

2. Modify and save the DeleteStacksetsStackParams file. For example, you can replace the contents with something like this:

```
{
   "Name": "Stackset name",
   "Region": "us-east-1",
   "Priority": "High"
 }
```

3. Output the RFC template JSON file to a file; this example names it DeleteStacksetsStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteStacksetsStackRfc.json
```

4. Modify and save the DeleteStacksetsStackRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1yqy4frl5s8y8",
  "Title": "Delete StackSets Stack"
}
```

5. Create the RFC, specifying the DeleteStacksetsStack Rfc file and the DeleteStacksetsStackParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteStacksetsStackRfc.json  --
execution-parameters file://DeleteStacksetsStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For AWS CloudFormation details, see Delete a stack set

- For general AWS CloudFormation information on stack sets, see StackSets concepts
- To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1yqy4frl5s8y8.

Example: Required Parameters

```
{
  "Name": "test-stackset"
}
```

Example: All Parameters

```
{
  "Name": "test-stackset",
  "Region": "us-east-1",
  "Priority": "High"
}
```

Management Account | Enable Developer Mode

Enable Developer Mode for an existing application account. Note that, in Developer mode, you are responsible for monitoring infrastructure resources that are provisioned outside of the AMS change management process.

Full classification: Management | Managed landing zone | Management account | Enable developer mode

Change Type Details

Change type ID	ct-1opjmhuddw194
Current version	1.0
Expected execution duration	3600 minutes

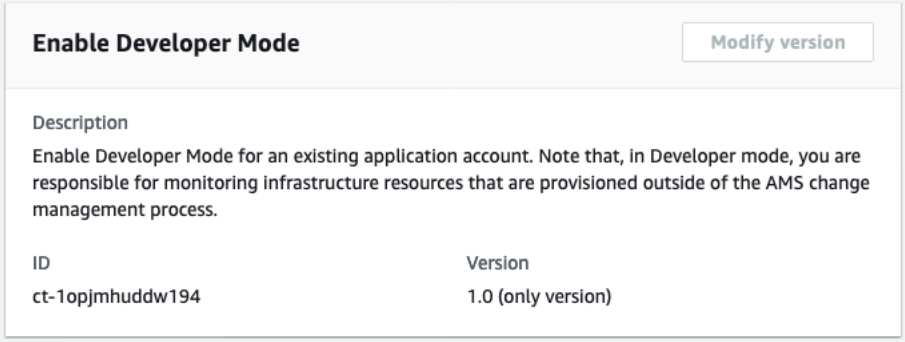
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable Developer mode

Management account: Enabling Management account Developer mode with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Enabling Management account Developer mode with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1opjmhuddw194" \
--change-type-version "1.0" --title "Enable developer mode" \
--execution-parameters "{\"ApplicationAccountId\": \"ACCOUNT_ID\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it EnableDevModeParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1opjmhuddw194" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > EnableDevModeParams.json
   ```

2. Modify and save the EnableDevModeParams file. For example, you can replace the contents with something like this:

   ```
   {
   "ApplicationAccountId":"ACCOUNT_ID"
   }
   ```

3. Output the RFC template JSON file to a file; this example names it EnableDevModeRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableDevModeRfc.json
   ```

4. Modify and save the EnableDevModeRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeId": "ct-1opjmhuddw194",
  "ChangeTypeVersion": "1.0",
  "Title": "Enable developer mode"
}
```

5. Create the RFC, specifying the EnableDevMode Rfc file and the EnableDevModeParams file:

```
aws amscm create-rfc --cli-input-json file://EnableDevModeRfc.json  --execution-
parameters file://EnableDevModeParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about developer mode, see Developer mode.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1opjmhuddw194.

Example: Required Parameters

```
{
  "ApplicationAccountId": "123456789012"
}
```

Example: All Parameters

```
Example not available.
```

Management Account | Move Account To OU

Move an account under an AWS organizational unit (OU) to a different OU.

Full classification: Management | Managed landing zone | Management account | Move account to OU

Change Type Details

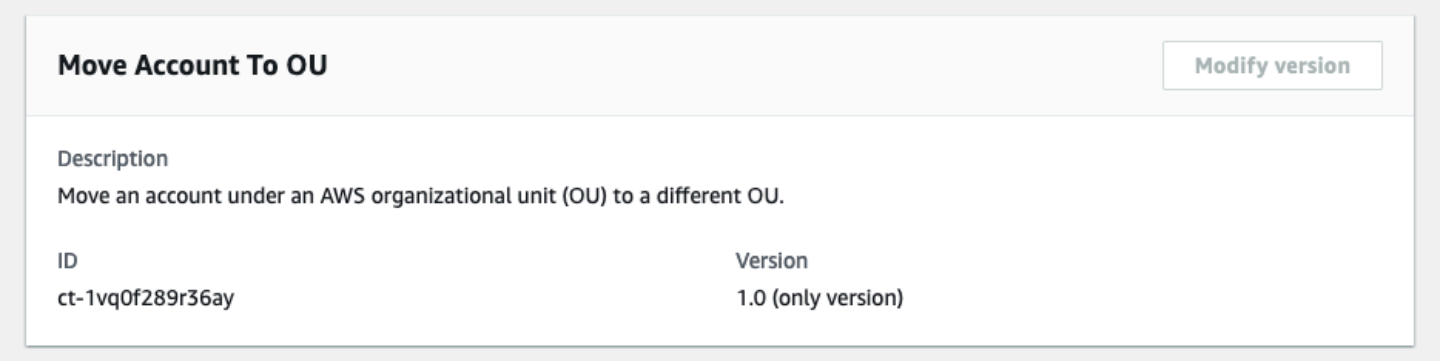
Change type ID	ct-1vq0f289r36ay
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Move account to OU

Moving an account to a different OU with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Moving an account to a different OU with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Note

Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-1vq0f289r36ay" \
--change-type-version "1.0" --title "Move Account To OU" \
--execution-parameters "{ \"AccountId\": \"ACCOUNT_ID\", \"TargetOUPath\":
 \"applications:managed:OU1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it MvAcctToOuParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1vq0f289r36ay" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > MvAcctToOuParams.json
   ```

2. Modify and save the MvAcctToOuParams file. For example, you can replace the contents with something like this:

```
{
    "AccountId": "ACCOUNT_ID",
    "TargetOUPath": "applications:managed:OU1",
}
```

3. Output the RFC template JSON file to a file; this example names it MvAcctToOuRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > MvAcctToOuRfc.json
```

4. Modify and save the MvAcctToOuRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-1vq0f289r36ay",
"Title":                "Move-Acct-To-OU-RFC"
}
```

5. Create the RFC, specifying the MvAcctToOuRfc file and the MvAcctToOuParams file:

```
aws amscm create-rfc --cli-input-json file://MvAcctToOuRfc.json  --execution-
parameters file://MvAcctToOuParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> When moving accounts between OUs having custom SCPs, functionality may break due to SCPs being applied or a security posture compromised due to SCPs being removed.
> When moving accounts from an OU having a custom stackset (with CloudFormation auto-deployment feature enabled (see Manage automatic deployments for a stack set with service-managed permissions, to an OU which does not have this stackset, AWS CloudFormation would remove the stackset instance from the account. This may cause

functionality loss. Vice versa, you should be aware of the addition of unwanted stackset instances when moving to a new OU.
When an account is moved, it may no longer have access to resources specified by OU based policy conditions (aws:PrincipalOrgID) in IAM/S3, (see AWS global condition context keys.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1vq0f289r36ay.

Example: Required Parameters

```
{
    "AccountId": "123456789012",
    "TargetOUPath": "applications:development"
}
```

Example: All Parameters

```
Example not available.
```

Management Account | Offboard Application Account

Offboard the specified application account. Run this from the management account for the application account that you want offboarded. You must first confirm the offboarding request by submitting the Confirm offboarding CT (ct-2wlfo2jxj2rkj) from the application account. If you are offboarding a customer-managed account, then ct-2wlfo2jxj2rkj is not needed. Only use these CTs when you plan to terminate all resources within the specified account and close the account. After you successfully submit both CTs, AMS can't undo the offboarding, repurpose the account, or help you to remediate issues in the account.

Full classification: Management | Managed landing zone | Management account | Offboard application account

Change Type Details

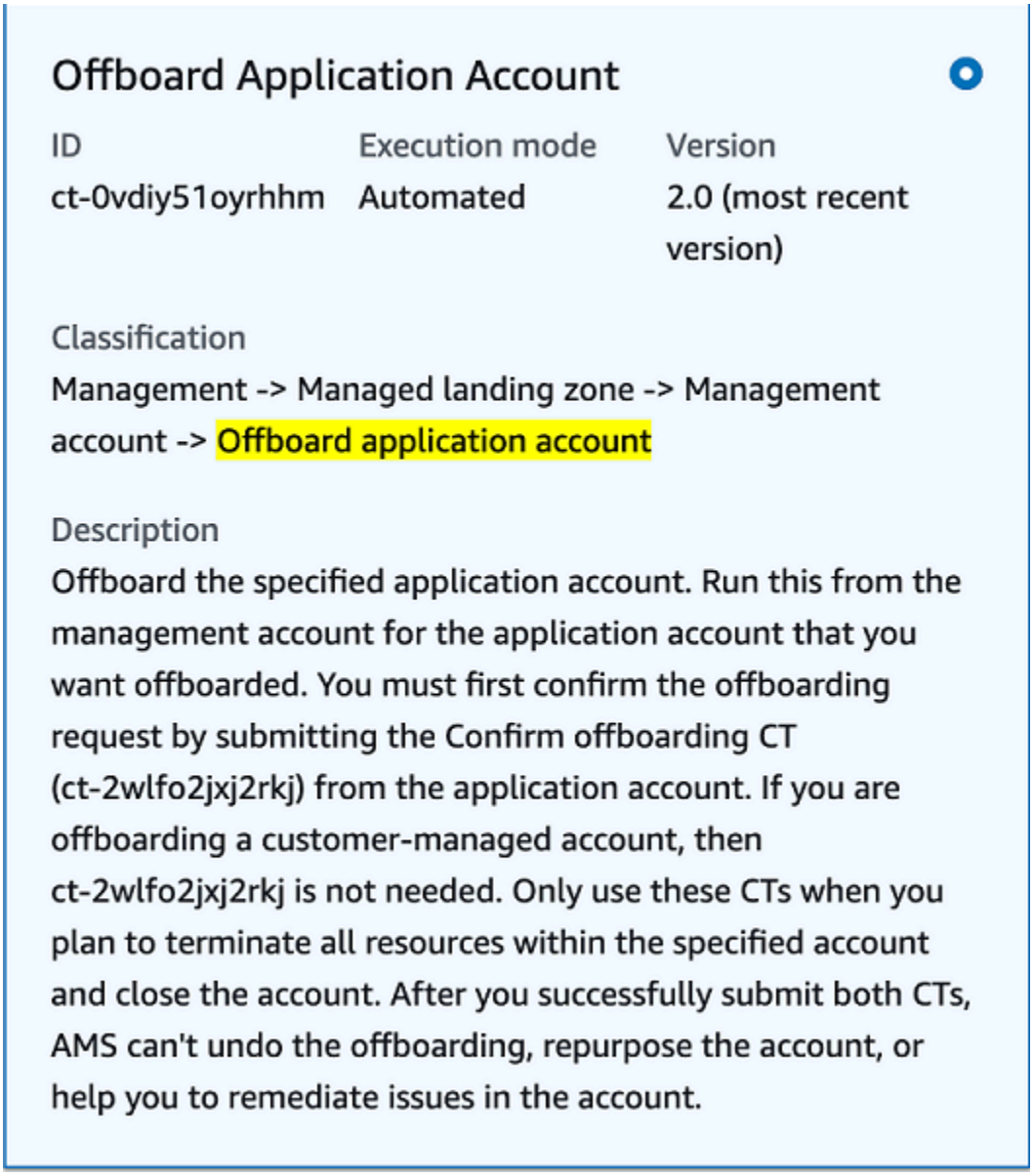
Change type ID	ct-0vdiy51oyrhhm
Current version	2.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Management account: Offboard Application account

Management account: Offboarding an Application account with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Management account: Offboarding an Application account with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from the Management account associated with the application account being offboarded.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0vdiy51oyrhhm" --change-type-version
 "2.0" --title "Run Offboarding" --execution-parameters "{\"AccountID\":
 \"000000000000\",\"AccountEmail\": \"email@amazon.com\",\"Confirmation\": \"confirm\",
\"DeleteTransitGatewayAttachment\":true}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RunAppAcctOffBParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0vdiy51oyrhhm" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > RunAppAcctOffBParams.json
   ```

2. Modify and save the RunAppAcctOffBParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "AccountID": "000000000000",
  "AccountEmail": "email@amazon.com",
  "Confirmation": "confirm",
  "DeleteTransitGatewayAttachment" : true
}
```

3. Output the RFC template JSON file to a file; this example names it RunAppAcctOffBRfc.json:

```
    aws amscm create-rfc --generate-cli-skeleton > RunAppAcctOffBRfc.json
```

4. Modify and save the RunAppAcctOffBRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "2.0",
  "ChangeTypeId": "ct-0vdiy51oyrhhm",
  "Title": "Execute Offboarding"
}
```

5. Create the RFC, specifying the RunAppAcctOffBRfc file and the RunAppAcctOffBParams file:

```
    aws amscm create-rfc --cli-input-json file://RunAppAcctOffBRfc.json  --
execution-parameters file://RunAppAcctOffBParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- First step to offboarding the AMS multi-account landing zone Application account is to submit the Confirm application account offboarding CT (ct-2wlfo2jxj2rkj) from the application account.
- There is no prerequisite or confirmation CT for Customer Managed application accounts.
- Note that offboarding is irreversible.
- If you intend to self-operate the account after offboarding from AMS, then make sure to specify `DeleteTransitGatewayAttachment` parameter as `false` to retain connectivity.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0vdiy51oyrhhm.

Example: Required Parameters

```
{
  "RequestType": "OffboardingExecution",
  "Parameters": {
    "AccountId": "000000000000",
    "AccountEmail": "example@email.com",
    "Confirmation": "confirm",
    "DeleteTransitGatewayAttachment": true
  }
}
```

Example: All Parameters

```
{
  "RequestType": "OffboardingExecution",
  "Parameters": {
    "AccountId": "000000000000",
    "AccountEmail": "example@email.com",
    "Confirmation": "confirm",
    "DeleteTransitGatewayAttachment": true
  }
}
```

Management Account | Update StackSets Stack (Review Required)

Update an existing AWS CloudFormation (CFN) StackSets stack to deploy, or to update, the instances of the stack.

Full classification: Management | Managed landing zone | Management account | Update StackSets stack (review required)

Change Type Details

Change type ID	ct-1v9g9n30woc8h

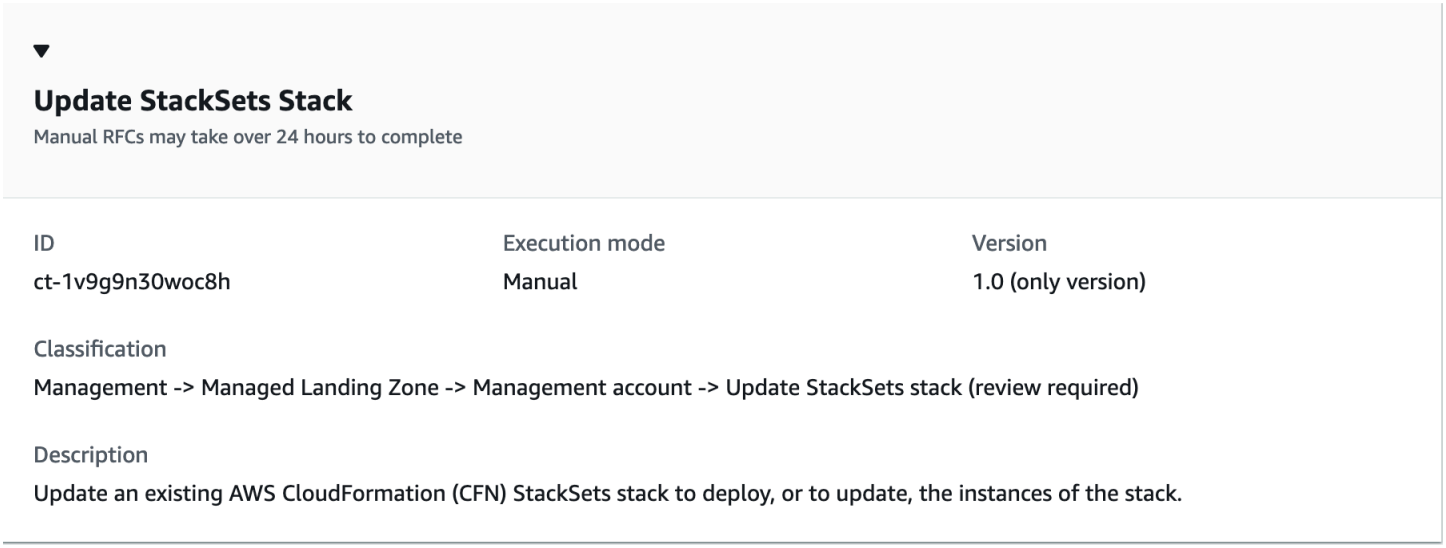
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update a Stacksets stack

Updating a Stacksets stack with the console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a Stacksets stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

> **Note**
>
> Run this change type from your Management account.

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1v9g9n30woc8h" --change-type-version "1.0"
 --title "Update StackSets Stack" --execution-parameters "{\"Name\": \"Stackset name\",
 \"Region\": \"us-east-1\", \"OuId\": \"ou-cccc-00000000\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateStacksetsStackParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1v9g9n30woc8h"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateStacksetsStackParams.json
   ```

2. Modify and save the UpdateStacksetsStackParams file. For example, you can replace the contents with something like this:

   ```
   {
      "CloudFormationTemplate": "CFN Template",
   ```

```
   "CloudFormationTemplateS3Endpoint": "S3 link to the template",
   "Description": "Update Test-Stackset",
   "Name": "test-stackset",
   "OuId": ["ou-cccc-00000000"],
   "Region": "us-east-1",
   "Parameters": [
     { "Name": "test-value",
       "Value": "test-value" }
   ],
   "Tags": [
     {
       "Key": "key1",
       "Value": "value1"
     },
     {
       "Key": "key2",
       "Value": "value2"
     }
    ],
    "Priority": "High"
}
```

3. Output the RFC template JSON file to a file; this example names it UpdateStacksetsStackRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateStacksetsStackRfc.json
```

4. Modify and save the UpdateStacksetsStackRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-1v9g9n30woc8h",
  "Title": "Update StackSets Stack"
}
```

5. Create the RFC, specifying the UpdateStacksetsStack Rfc file and the UpdateStacksetsStackParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateStacksetsStackRfc.json  --
execution-parameters file://UpdateStacksetsStackParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

- For AWS CloudFormation details, see Create a stack set
- For general AWS CloudFormation information on stack sets, see StackSets concepts
- To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1v9g9n30woc8h.

Example: Required Parameters

```
{
  "Name": "test-stackset",
  "OuId": ["ou-cccc-00000000"],
  "Region": "us-east-1",
  "Priority": "High"
}
```

Example: All Parameters

```
{
  "CloudFormationTemplate": "template",
  "CloudFormationTemplateS3Endpoint": "https://s3.amazonaws.com/cf-
templates-33kj7hiuwdk9-us-east-1/2017261mYA-stm-dynamic-sqs-no-params-
sept-2017.template",
  "Description": "AMSTestCT - Update Test-Stackset",
  "Name": "test-stackset",
  "OuId": ["ou-cccc-00000000"],
  "Region": "us-east-1",
  "Parameters": [
    { "Name": "test-value",
      "Value": "test-value" }
  ],
```

```
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    }
  ],
  "Priority": "High"
}
```

Networking Account | Associate TGW Attachment

Associate transit gateway (TGW) attachment to the transit gateway (TGW) route table. Use this change type for multi-account landing zone (MALZ) in Networking account only.

Full classification: Management | Managed landing zone | Networking account | Associate TGW attachment

Change Type Details

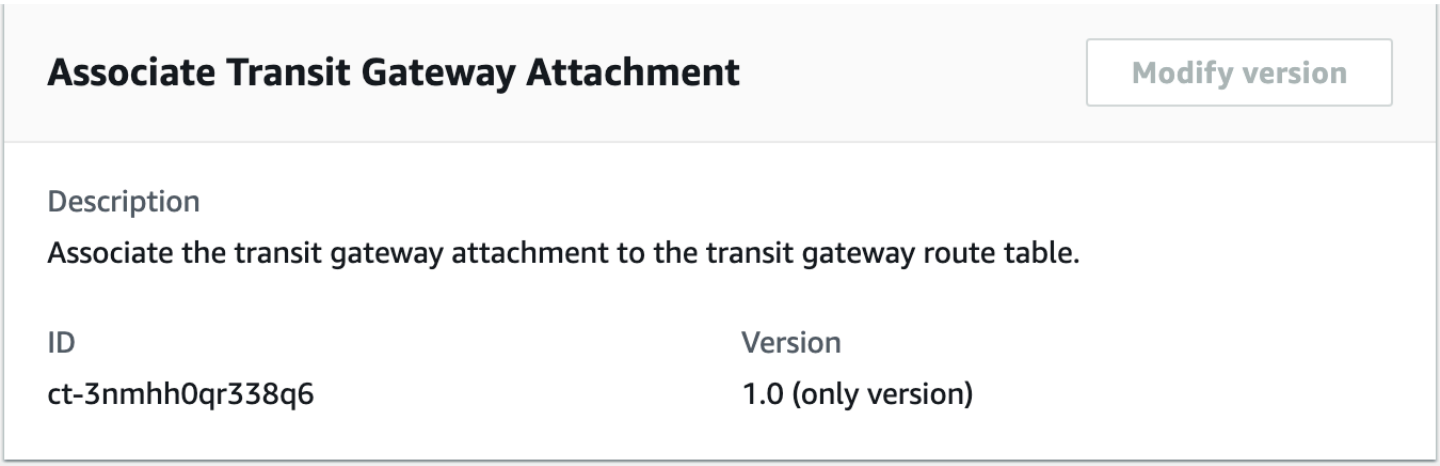
Change type ID	ct-3nmhh0qr338q6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Associate TGW attachment

Networking account: Associate a TGW attachment with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Associate a TGW attachment with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3nmhh0qr338q6" --change-type-version "1.0"
 --title "Associate Transit Gateway Attachment" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-AssociateTGWAttachment\",\"Region\": \"us-east-1\",
\"Parameters\": {\"TransitGatewayAttachmentId\": [\"tgw-attach-0878cf82a40721d19\"],
\"TransitGatewayRouteTableId\": [\"tgw-rtb-06ddc751c0c0c881c\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it AssociateTgwAttachmentParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3nmhh0qr338q6"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    AssociateTgwAttachmentParams.json
   ```

2. Modify and save the AssociateTgwAttachmentParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-AssociateTGWAttachment",
     "Region": "us-east-1",
     "Parameters": {
       "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
       "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it AssociateTgwAttachmentRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > AssociateTgwAttachmentRfc.json
   ```

4. Modify and save the AssociateTgwAttachmentRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-3nmhh0qr338q6",
     "Title": "Associate Transit Gateway Attachment"
   }
   ```

5. Create the RFC, specifying the AssociateTgwAttachmentRfc file and the AssociateTgwAttachmentParams file:

```
aws amscm create-rfc --cli-input-json file://AssociateTgwAttachmentRfc.json  --
execution-parameters file://AssociateTgwAttachmentParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3nmhh0qr338q6.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-AssociateTGWAttachment",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-AssociateTGWAttachment",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Networking Account | Disable TGW Propagation

Disable the Transit Gateway (TGW) attachment from propagating routes to the TGW route table. For multi-account landing zone (MALZ), use this change type in the Network account only.

Full classification: Management | Managed landing zone | Networking account | Disable TGW propagation

Change Type Details

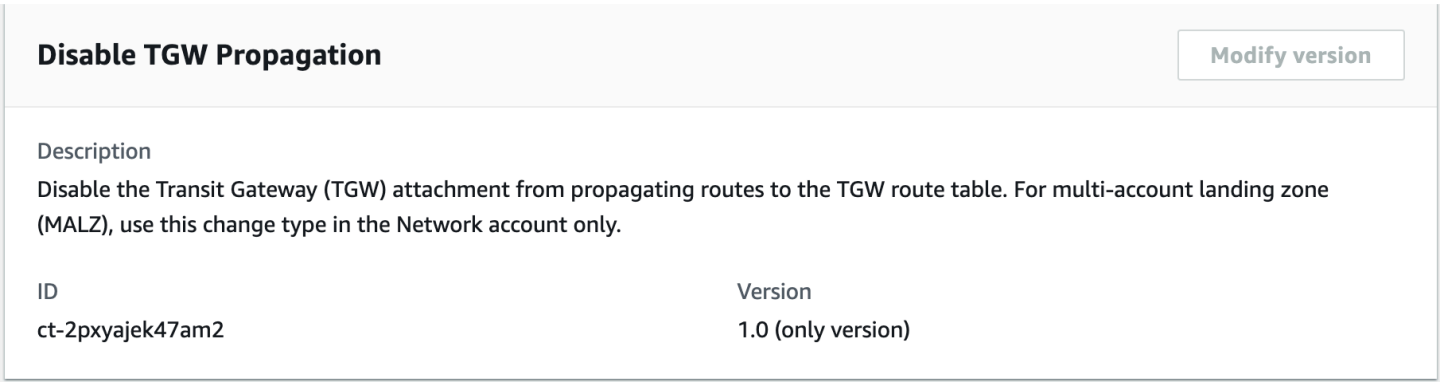
Change type ID	ct-2pxyajek47am2
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Disable TGW propagation

Networking account: Disable TGW propagation with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Disable TGW propagation with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2pxyajek47am2" --change-type-version "1.0"
 --title "Disable Transit Gateway Propagation" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-DisableTGWRouteTablePropagation\",\"Region\": \"us-east-1\",
\"Parameters\": {\"TransitGatewayAttachmentId\": [\"tgw-attach-01234567890abcdef\"],
\"TransitGatewayRouteTableId\": [\"tgw-rtb-01234567890abcdef\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it TgwPropagationDisableParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2pxyajek47am2"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TgwPropagationDisableParams.json
   ```

2. Modify and save the TgwPropagationDisableParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DisableTGWRouteTablePropagation",
     "Region": "us-east-1",
     "Parameters": {
       "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
       "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it TgwPropagationDisableRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > TgwPropagationDisableRfc.json
   ```

4. Modify and save the TgwPropagationDisableRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-2pxyajek47am2",
     "Title": "Disable Transit Gateway Propagation"
   }
   ```

5. Create the RFC, specifying the TgwPropagationDisableRfc file and the TgwPropagationDisableParams file:

```
aws amscm create-rfc --cli-input-json file://TgwPropagationDisableRfc.json  --
execution-parameters file://TgwPropagationDisableParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2pxyajek47am2.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DisableTGWRouteTablePropagation",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DisableTGWRouteTablePropagation",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
  }
}
```

Networking Account | Disassociate TGW Attachment

Disassociate transit gateway (TGW) attachment from the transit gateway (TGW) route table. Use this change type for multi-account landing zone (MALZ) in Networking account only.

Full classification: Management | Managed landing zone | Networking account | Disassociate TGW attachment

Change Type Details

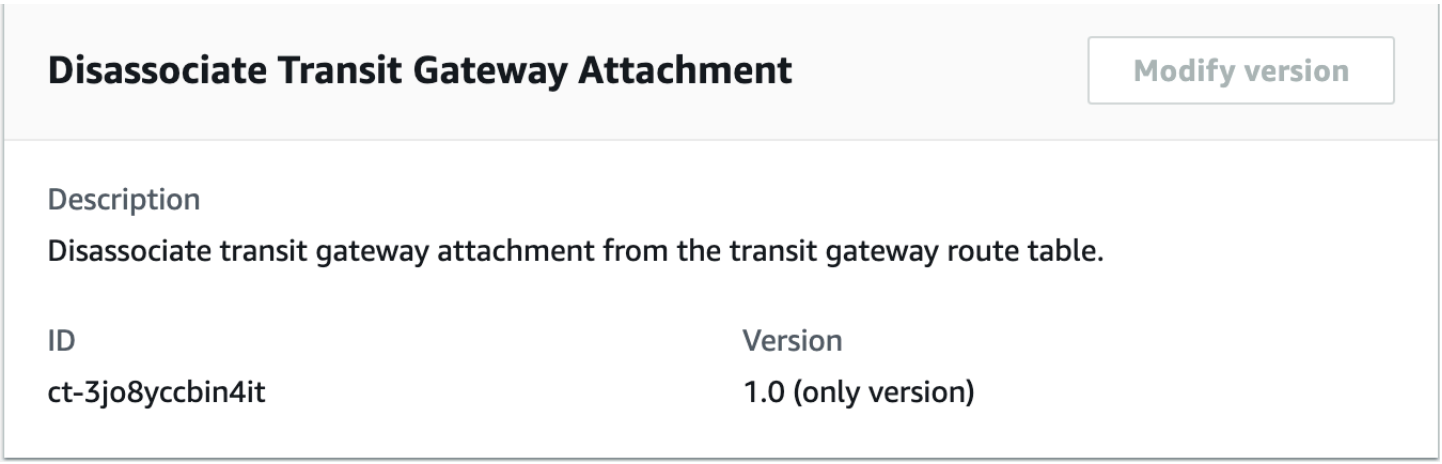
Change type ID	ct-3jo8yccbin4it
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Disassociate TGW attachment

Networking account: Disassociate a TGW attachment with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Disassociate a TGW attachment with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3jo8yccbin4it" --change-type-version "1.0"
 --title "Disassociate a TGW attachment" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-CreateRouteInTGWRouteTable\",\"Region\": \"us-east-1\",
\"Parameters\": {\"TransitGatewayAttachmentId\": [\"tgw-attach-0878cf82a40721d19\"],
\"TransitGatewayRouteTableId\": [\"tgw-rtb-06ddc751c0c0c881c\"], \"Blackhole\":
 [\"false\"], \"DestinationCidrBlock\": [\"10.0.0.0/24\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DisassociateTgwAttachmentParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3jo8yccbin4it"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DisassociateTgwAttachmentParams.json
   ```

2. Modify and save the DisassociateTgwAttachmentParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-DisassociateTGWAttachment",
     "Region": "us-east-1",
     "Parameters": {
       "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
       "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it DisassociateTgwAttachmentRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DisassociateTgwAttachmentRfc.json
   ```

4. Modify and save the DisassociateTgwAttachmentRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-3jo8yccbin4it",
     "Title": "Disassociate a TGW attachment"
   }}
   ```

5. Create the RFC, specifying the DisassociateTgwAttachmentRfc file and the DisassociateTgwAttachmentParams file:

```
aws amscm create-rfc --cli-input-json file://DisassociateTgwAttachmentRfc.json  --
execution-parameters file://DisassociateTgwAttachmentParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3jo8yccbin4it.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DisassociateTGWAttachment",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DisassociateTGWAttachment",
```

```
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-0878cf82a40721d19" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-06ddc751c0c0c881c" ]
  }
}
```

Networking Account | Enable TGW Propagation

Enable the Transit Gateway (TGW) attachment to propagate routes to the TGW route table. For multi-account landing zone (MALZ), use this change type in the Network account only.

Full classification: Management | Managed landing zone | Networking account | Enable TGW propagation

Change Type Details

Change type ID	ct-1f9hi4bephqa9
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable TGW propagation

Networking account: enable TGW propagation with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Enable TGW propagation with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1f9hi4bephqa9" --change-type-version "1.0"
 --title "Enable Transit Gateway Propagation" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-EnableTGWRouteTablePropagation\",\"Region\": \"us-east-1\",
\"Parameters\": {\"TransitGatewayAttachmentId\": [\"tgw-attach-01234567890abcdef\"],
\"TransitGatewayRouteTableId\": [\"tgw-rtb-01234567890abcdef\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it TgwPropagationEnableParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-1f9hi4bephqa9"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TgwPropagationEnableParams.json
   ```

2. Modify and save the TgwPropagationEnableParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-EnableTGWRouteTablePropagation",
     "Region": "us-east-1",
     "Parameters": {
       "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
       "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
     }
   }
   ```

3. Output the RFC template JSON file to a file; this example names it TgwPropagationEnableRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > TgwPropagationEnableRfc.json
   ```

4. Modify and save the TgwPropagationEnableRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
     "ChangeTypeVersion": "1.0",
     "ChangeTypeId": "ct-1f9hi4bephqa9",
     "Title": "Enable Transit Gateway Propagation"
   }
   ```

5. Create the RFC, specifying the TgwPropagationEnableRfc file and the TgwPropagationEnableParams file:

```
aws amscm create-rfc --cli-input-json file://TgwPropagationEnableRfc.json  --
execution-parameters file://TgwPropagationEnableParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zone, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1f9hi4bephqa9.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-EnableTGWRouteTablePropagation",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-EnableTGWRouteTablePropagation",
  "Region": "us-east-1",
```

```
  "Parameters": {
    "TransitGatewayAttachmentId": [ "tgw-attach-01234567890abcdef" ],
    "TransitGatewayRouteTableId": [ "tgw-rtb-01234567890abcdef" ]
  }
}
```

Networking Account | Remove TGW Static Route

Remove the specified TGW static route from the specified transit gateway (TGW) route table. Use this multi-account landing zone (MALZ) change type only in a Networking account.

Full classification: Management | Managed landing zone | Networking account | Remove TGW static route

Change Type Details

Change type ID	ct-0rmgrnr9w8mzh
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remove TGW static route

Networking account: Remove a TGW static route with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Networking account: Remove a TGW static route with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0rmgrnr9w8mzh" --change-type-version
 "1.0" --title "Remove TGW Static Route" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-RemoveRouteFromTGWRouteTable\",\"Region\": \"us-east-1\",
\"Parameters\": {\"TransitGatewayRouteTableId\": \"tgw-rtb-06ddc751c0c0c881c\",
 \"DestinationCidrBlock\": \"10.16.1.0/24\"}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemoveTgwStaticRouteParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0rmgrnr9w8mzh"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RemoveTgwStaticRouteParams.json
   ```

2. Modify and save the RemoveTgwStaticRouteParams file. For example, you can replace the contents with something like this:

   ```
   {
   ```

```
  "DocumentName": "AWSManagedServices-RemoveRouteFromTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayRouteTableId": "tgw-rtb-06ddc751c0c0c881c",
    "DestinationCidrBlock": "10.16.1.0/24"
  }
}
```

3. Output the RFC template JSON file to a file; this example names it RemoveTgwStaticRouteRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RemoveTgwStaticRouteRfc.json
```

4. Modify and save the RemoveTgwStaticRouteRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-0rmgrnr9w8mzh",
  "Title": "Remove TGW Static Route"
}
```

5. Create the RFC, specifying the RemoveTgwStaticRouteRfc file and the RemoveTgwStaticRouteParams file:

```
aws amscm create-rfc --cli-input-json file://RemoveTgwStaticRouteRfc.json  --
execution-parameters file://RemoveTgwStaticRouteParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This Change Type is only valid in Multi-account Landing Zone (MALZ) Networking accounts.

To learn more about AMS multi-account landing zones, see AWS Managed Services (AMS) Now Offers Managed Landing Zones.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0rmgrnr9w8mzh.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-RemoveRouteFromTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayRouteTableId": ["tgw-rtb-06ddc751c0c0c881c"],
    "DestinationCidrBlock": ["10.16.1.0/24"]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-RemoveRouteFromTGWRouteTable",
  "Region": "us-east-1",
  "Parameters": {
    "TransitGatewayRouteTableId": ["tgw-rtb-06ddc751c0c0c881c"],
    "DestinationCidrBlock": ["10.16.1.0/24"]
  }
}
```

Monitoring and Notification Subcategory

Change Type Items and Operations in the Monitoring and Notification Subcategory

- CloudWatch | Enable Non-Root Volumes Monitoring
- GuardDuty IP Set | Delete (Review Required)
- GuardDuty IP Set | Update (Review Required)
- GuardDuty Threat Intel Set | Delete (Review Required)
- GuardDuty Threat Intel Set | Update (Review Required)
- SNS | Subscribe To DirectCustomerAlerts
- SQS | Update

CloudWatch | Enable Non-Root Volumes Monitoring

Enable monitoring on non-root volumes of an EC2 instance.

Full classification: Management | Monitoring and notification | CloudWatch | Enable Non-Root Volumes Monitoring

Change Type Details

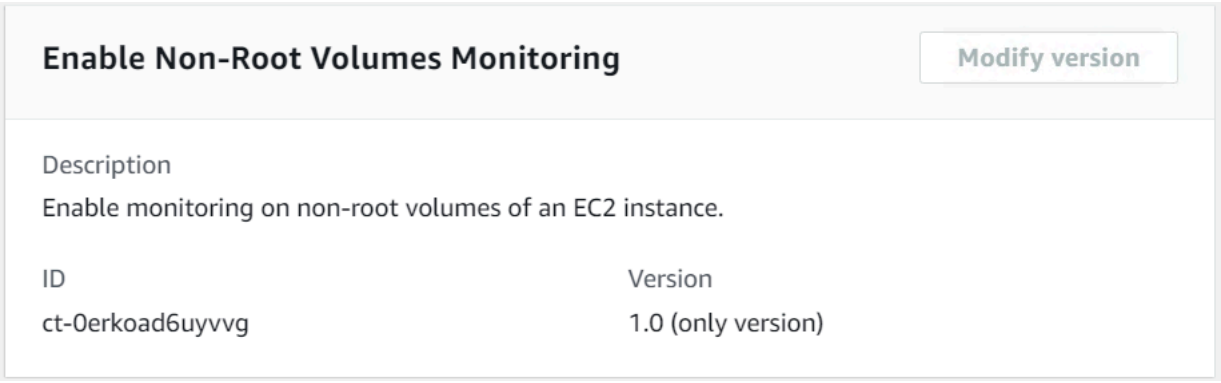
Change type ID	ct-0erkoad6uyvvg
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Enable CloudWatch non-root volumes monitoring

Enabling CloudWatch non-root volume monitoring with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Enabling CloudWatch non-root volume monitoring with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

Monitor the RFC: `aws amscm get-rfc --rfc-id ID` command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0erkoad6uyvvg" --change-type-version "1.0"
 --title "Enable Non-Root Volumes Monitoring" --execution-parameters "{\"DocumentName
\":\"AWSManagedServices-DeployNonRootVolumeMonitoring\",\"Region\":\"us-east-1\",
\"Parameters\":{\"InstanceId\":[\"i-1234567890abcdef0\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file in your current folder; this example names it CwNonRootVolumeMonitoringParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-0erkoad6uyvvg"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    CwNonRootVolumeMonitoringParams.json
   ```

2. Modify and save the CwNonRootVolumeMonitoringParams.json file. For example, you can replace the contents with something like this:

```
{
    "DocumentName": "AWSManagedServices-DeployNonRootVolumeMonitoring",
    "Region": "us-east-1",
    "Parameters": {
        "InstanceId": [
            "i-1234567890abcdef0"
        ]
    }
}
```

3. Output the JSON template for CreateRfc to a file in your current folder; this example names it CwNonRootVolumeMonitoringRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > CwNonRootVolumeMonitoringRfc.json
```

4. Modify and save the CwNonRootVolumeMonitoringRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0erkoad6uyvvg",
"Title":                "CW-NON-ROOT-VOL-MONITORING-RFC"
}
```

5. Create the RFC, specifying the CwNonRootVolumeMonitoringRfc file and the execution parameters file:

```
aws amscm create-rfc --cli-input-json file://CwNonRootVolumeMonitoringRfc.json --
execution-parameters file://CwNonRootVolumeMonitoringParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about CloudWatch, see Enable or disable detailed monitoring for your instances.

The EC2 instance alart `Non-root volume usage` is **DISABLED** by default. If you require alert generation based on this alarm, then you must enable it using this RFC.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0erkoad6uyvvg.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-DeployNonRootVolumeMonitoring",
  "Region": "us-east-1",
  "Parameters": {
    "InstanceId": [
      "i-1234567890abcdef0"
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-DeployNonRootVolumeMonitoring",
  "Region": "us-east-1",
  "Parameters": {
    "InstanceId": [
      "i-1234567890abcdef0"
    ]
  }
}
```

GuardDuty IP Set | Delete (Review Required)

Use to delete an Amazon GuardDuty IPSet instance which is a list of trusted IP addresses that have been whitelisted for highly secure communication with your AWS environment.

Full classification: Management | Monitoring and notification | GuardDuty IP set | Delete (review required)

Change Type Details

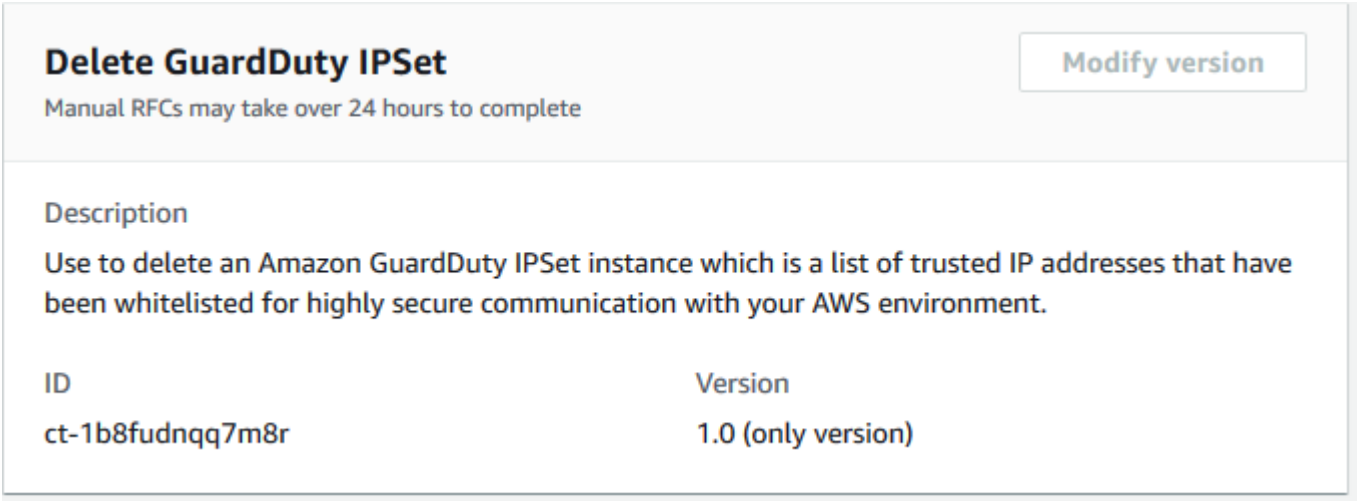
Change type ID	ct-1b8fudnqq7m8r
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete GuardDuty IP set (review required)

Deleting an IP set for GuardDuty (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting an IP set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1b8fudnqq7m8r" --change-type-
version "1.0" --title "Delete Amazon GuardDuty IP Set" --execution-parameters
 "{\"DetectorId\": \"00000000000000000000000000000000\", \"IpSetId\":
 \"00000000000000000000000000000000\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteGdIpSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-08avsj2e9mc7g" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > DeleteGdIpSetParams.json
   ```

2. Modify and save the DeleteGdIpSetParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DetectorId": "00000000000000000000000000000000",
     "IpSetId": "00000000000000000000000000000000",
     "Region": "us-east-1"
   ```

```
}
```

3. Output the RFC template JSON file to a file named DeleteGdIpSetRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > DeleteGdIpSetRfc.json
```

4. Modify and save the DeleteGdIpSetRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-08avsj2e9mc7g",
"Title":                "DELETE_GD_IP_SET"
}
```

5. Create the RFC, specifying the DeleteGdIpSetRfc Rfc file and the DeleteGdIpSetParams file:

```
aws amscm create-rfc --cli-input-json file://DeleteGdIpSetRfc.json  --execution-
parameters file://DeleteGdIpSetParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For more information about Amazon GuardDuty, see Amazon GuardDuty.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1b8fudnqq7m8r.

Example: Required Parameters

```
{
```

```
  "IpSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Region": "us-east-1"
}
```

Example: All Parameters

```
{
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "IpSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Region": "us-east-1",
  "Priority": "Medium"
}
```

GuardDuty IP Set | Update (Review Required)

Use to update an Amazon GuardDuty IPSet instance which is a list of trusted IP addresses that have been whitelisted for highly secure communication with your AWS environment.

Full classification: Management | Monitoring and notification | GuardDuty IP set | Update (review required)

Change Type Details

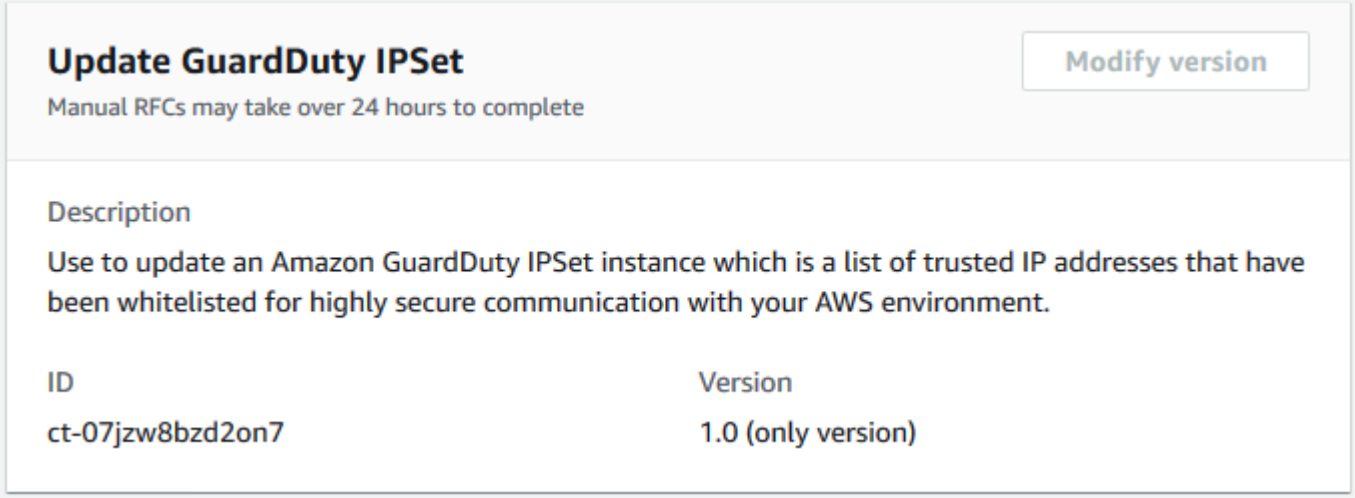
Change type ID	ct-07jzw8bzd2on7
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update GuardDuty IP set (review required)

Updating an IP set for GuardDuty (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an IP set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-07jzw8bzd2on7" --change-type-version "1.0"
 --title "Update Amazon GuardDuty IP Set" --execution-parameters "{\"Activate\": true,
 \"DetectorId\": \"00000000000000000000000000000000\", \"Name\": \"trusted-ips\",
 \"IpSet\": \"https://s3.us-west-2.amazonaws.com/my-bucket/my-object-key\", \"IpSetId
\": \"00000000000000000000000000000000\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateGdIpSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-07jzw8bzd2on7" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > UpdateGdIpSetParams.json
   ```

2. Modify and save the UpdateGdIpSetParams file. For example, you can replace the contents with something like this:

   ```
   {
     "Activate": true,
     "DetectorId": "00000000000000000000000000000000",
     "Name": "trusted-ips",
     "IpSet": "https://s3.us-west-2.amazonaws.com/my-bucket/my-object-key",
     "IpSetId": "00000000000000000000000000000000",
     "Region": "us-east-1"
   }
   ```

3. Output the RFC template JSON file to a file named UpdateGdIpSetRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > UpdateGdIpSetRfc.json
   ```

4. Modify and save the UpdateGdIpSetRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   ```

```
"ChangeTypeId":         "ct-07jzw8bzd2on7",
"Title":                "UPDATE_GD_IP_SET"
}
```

5. Create the RFC, specifying the UpdateGdIpSet Rfc file and the UpdateGdIpSetParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateGdIpSetRfc.json  --execution-
parameters file://UpdateGdIpSetParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

For more information about Amazon GuardDuty and creating IP sets, see Amazon GuardDuty and CreateIPSet.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-07jzw8bzd2on7.

Example: Required Parameters

```
{
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "IpSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Region": "us-east-1"
}
```

Example: All Parameters

```
{
  "Activate": true,
```

```
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "IpSet": "https://s3.amazonaws.com/guarddutylists/sample.txt",
  "IpSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Name": "Sample IPSet",
  "Region": "us-east-1",
  "Priority": "Medium"
}
```

GuardDuty Threat Intel Set | Delete (Review Required)

Use to delete an Amazon GuardDuty ThreatIntelSet instance which is a list of known malicious IP addresses.

Full classification: Management | Monitoring and notification | GuardDuty threat intel set | Delete (review required)

Change Type Details

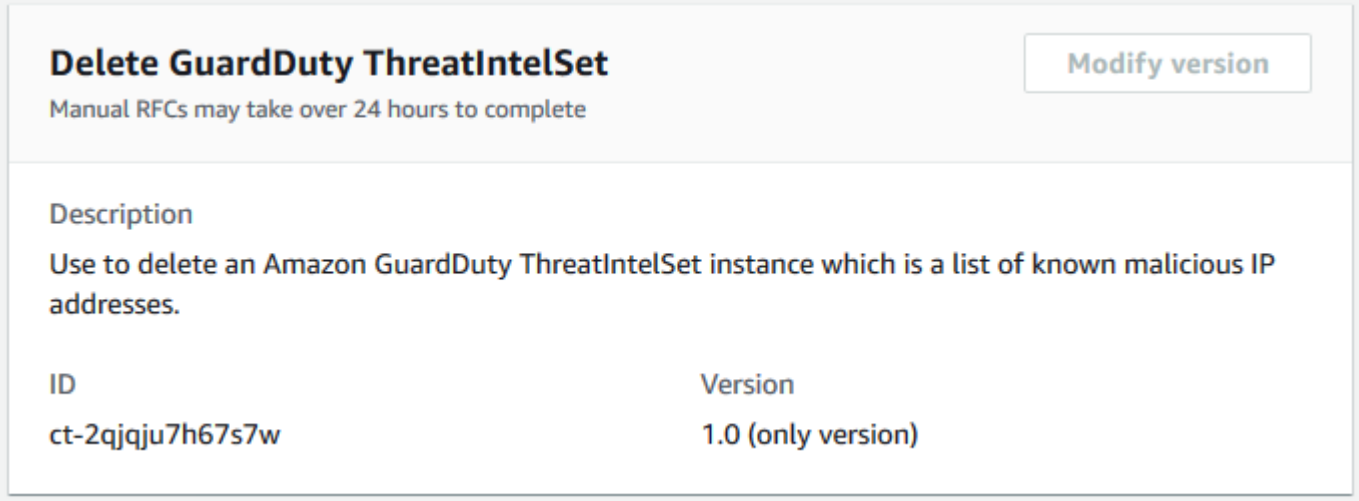
Change type ID	ct-2qjqju7h67s7w
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Delete GuardDuty Threat intel set (review required)

Deleting a Threat intel set for GuardDuty (review required) with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting a Threat intel set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2qjqju7h67s7w" --change-type-version
 "1.0" --title "Delete Amazon GuardDuty Threat Intel Set" --execution-parameters
 "{\"DetectorId\": \"00000000000000000000000000000000\", \"ThreatIntelSetId\":
 \"00000000000000000000000000000000\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it DeleteGdThreatIntelSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2qjqju7h67s7w"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    DeleteGdThreatIntelSetParams.json
   ```

2. Modify and save the DeleteGdThreatIntelSetParams file. For example, you can replace the contents with something like this:

   ```
   {
     "DetectorId": "00000000000000000000000000000000",
     "ThreatIntelSetId": "00000000000000000000000000000000",
     "Region": "us-east-1"
   }
   ```

3. Output the RFC template JSON file to a file named DeleteGdThreatIntelSetRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteGdThreatIntelSetRfc.json
   ```

4. Modify and save the DeleteGdThreatIntelSetRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-2qjqju7h67s7w",
   "Title":                "DELETE_GD_THREAT_INTEL_SET"
   }
   ```

5. Create the RFC, specifying the DeleteGdThreatIntelSet Rfc file and the DeleteGdThreatIntelSetParams file:

   ```
   aws amscm create-rfc --cli-input-json file://DeleteGdThreatIntelSetRfc.json  --
   execution-parameters file://DeleteGdThreatIntelSetParams.json
   ```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

For more information about Amazon GuardDuty and Threat Intel sets, see Amazon GuardDuty.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2qjqju7h67s7w.

Example: Required Parameters

```
{
  "Region": "us-east-1",
  "ThreatIntelSetId": "0cb0141ab9fbde177613ab9436212e90"
}
```

Example: All Parameters

```
{
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "Region": "us-east-1",
  "ThreatIntelSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Priority": "Medium"
}
```

GuardDuty Threat Intel Set | Update (Review Required)

Use to update an Amazon GuardDuty ThreatIntelSet instance which is a list of trusted IP addresses that have been whitelisted for highly secure communication with your AWS environment.

Full classification: Management | Monitoring and notification | GuardDuty threat intel set | Update (review required)

Change Type Details

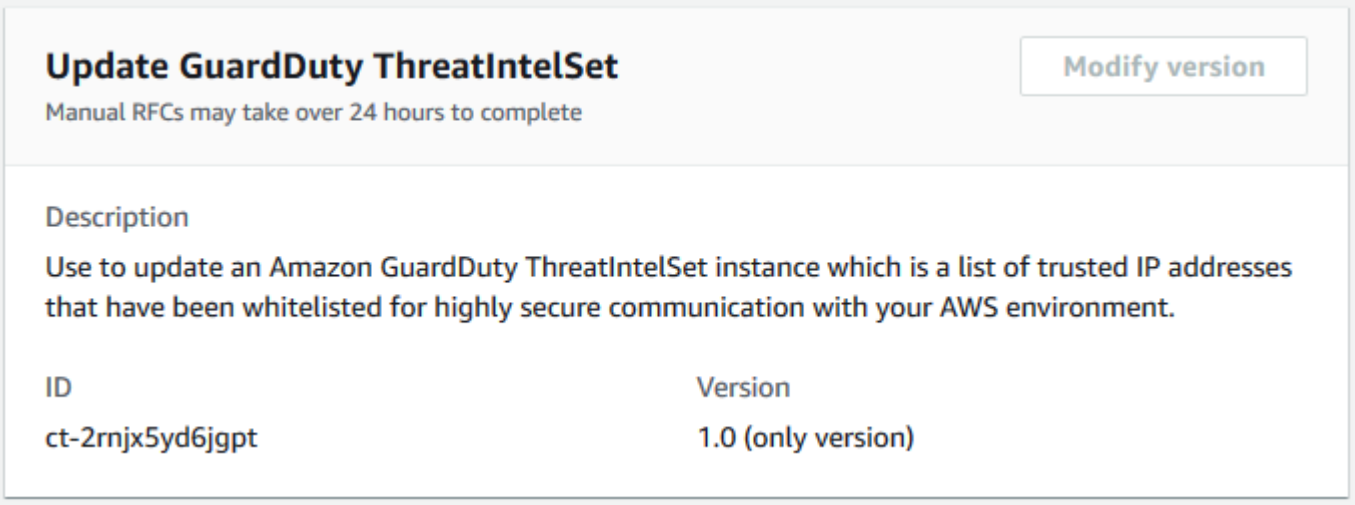
Change type ID	ct-2rnjx5yd6jgpt
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update GuardDuty Threat intel set (review required)

Updating a Threat intel set for GuardDuty (review required) with the console

THh following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a Threat intel set for GuardDuty (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2rnjx5yd6jgpt" --change-type-version
 "1.0" --title "Update Amazon GuardDuty Threat Intel Set" --execution-parameters
 "{\"Activate\": true, \"DetectorId\": \"00000000000000000000000000000000\", \"Name
\": \"blacklisted-ips\", \"ThreatIntelSet\": \"https://s3.us-west-2.amazonaws.com/my-
bucket/my-object-key\", \"ThreatIntelSetId\": \"00000000000000000000000000000000\",
 \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateGdThreatIntelSetParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-2rnjx5yd6jgpt"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdateGdThreatIntelSetParams.json
   ```

2. Modify and save the UpdateGdThreatIntelSetParams file. For example, you can replace the contents with something like this:

```
{
  "Activate": true,
  "DetectorId": "00000000000000000000000000000000",
  "Name": "blacklisted-ips",
  "ThreatIntelSet": "https://s3.us-west-2.amazonaws.com/my-bucket/my-object-key",
  "ThreatIntelSetId": "00000000000000000000000000000000",
  "Region": "us-east-1"
}
```

3. Output the RFC template JSON file to a file named UpdateGdThreatIntelSetRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateGdThreatIntelSetRfc.json
```

4. Modify and save the UpdateGdThreatIntelSetRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-2rnjx5yd6jgpt",
"Title":                "CREATE_GD_IP_SET"
}
```

5. Create the RFC, specifying the UpdateGdThreatIntelSet Rfc file and the UpdateGdThreatIntelSetParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateGdThreatIntelSetRfc.json  --
execution-parameters file://UpdateGdThreatIntelSetParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

For more information about Amazon GuardDuty and creating Threat Intel sets, see Amazon Guard Duty and CreateThreatIntelSet.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2rnjx5yd6jgpt.

Example: Required Parameters

```
{
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "ThreatIntelSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Region": "us-east-1"
}
```

Example: All Parameters

```
{
  "Activate": true,
  "DetectorId": "12abc34d567e8fa901bc2d34e56789f0",
  "ThreatIntelSet": "https://s3.amazonaws.com/guarddutylists/sample.txt",
  "ThreatIntelSetId": "0cb0141ab9fbde177613ab9436212e90",
  "Name": "Sample ThreatIntelSet",
  "Region": "us-east-1",
  "Priority": "Medium"
}
```

SNS | Subscribe To DirectCustomerAlerts

Subscribe an email address to the Direct-Customer-Alerts SNS topic.

Full classification: Management | Monitoring and notification | SNS | Subscribe to DirectCustomerAlerts

Change Type Details

Change type ID	ct-3rcl9u1k017wu
Current version	1.0

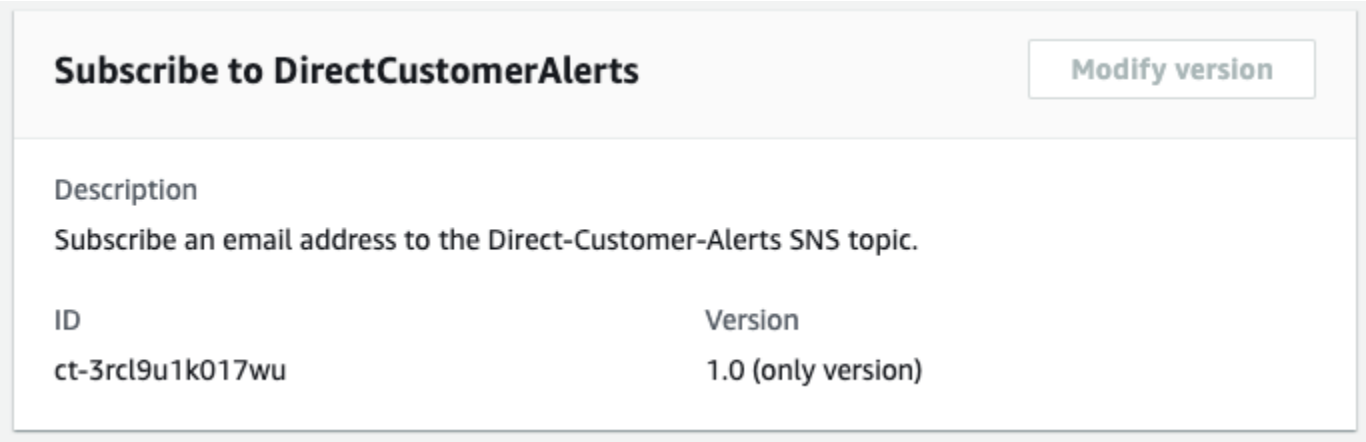
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Subscribe to SNS DirectCustomerAlerts

Subscribing to the Direct-Customer-Alerts SNS topic with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Subscribing to the Direct-Customer-Alerts SNS topic with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:?

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3rcl9u1k017wu" --change-type-version "1.0" --
title "Subscribe-Direct-Customer-Alerts" --execution-parameters "{\"Email\": \"sample-
email@example.com\", \"Region\": \"us-east-1\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it SnsSubscribeParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-3rcl9u1k017wu" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > SnsSubscribeParams.json
   ```

2. Modify and save the SnsSubscribeParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Description":      "SnsTopicSub-Create",
   "VpcId": "VPC_ID",
   "Name":             "My-SnsTopicSub",
   "Parameters":{
       "TopicName": "mytopic-cli-all-params",
       "DisplayName": "testsns",
       "Subscription1Protocol": "email",
       "Subscription1Endpoint": "abc@xyz.com",
       "Subscription1RawMessageDelivery": "false",
       "Subscription2Protocol": "sms",
       "Subscription2Endpoint": "+61500444777",
       "Subscription2RawMessageDelivery": "false",
   ```

```
    "KmsMasterKeyId": "arn:aws:kms:us-east-1:123456789101:key/
cfe0542d-3be9-4166-9eac-d0cd6af61445"
    }
}
```

3. Output the RFC template JSON file to a file named SnsSubscribeRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > SnsSubscribeRfc.json
```

4. Modify and save the SnsSubscribeRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3rcl9u1k017wu",
"Title":                "Subscribe-Direct-Customer-Alerts-RFC"
}
```

5. Create the RFC, specifying the SnsSubscribe Rfc file and the SnsSubscribeParams file:

```
aws amscm create-rfc --cli-input-json file://SnsSubscribeRfc.json  --execution-
parameters file://SnsSubscribeParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about AWS Simple Notification Service (SNS), see Amazon Simple Notification Service. Also see Getting Started with Amazon SNS.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3rcl9u1k017wu.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
Example not available.
```

SQS | Update

Use to modify the properties of an existing Amazon Simple Queue Service instance.

Full classification: Management | Monitoring and notification | SQS | Update

Change Type Details

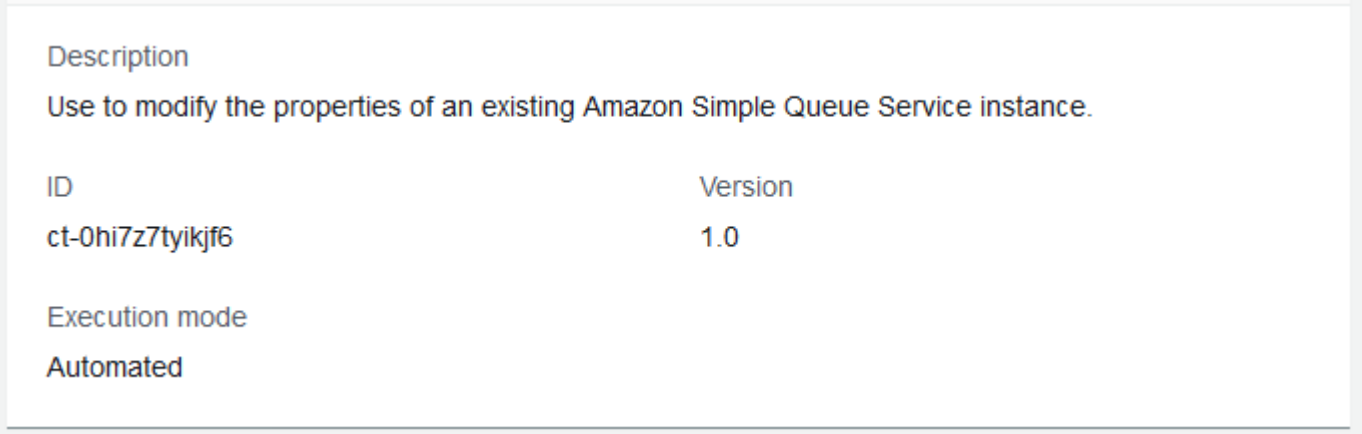
Change type ID	ct-0hi7z7tyikjf6
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update SQS queue

Updating an SQS queue with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an SQS queue with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0hi7z7tyikjf6" --change-type-version
 "1.0" --title "Update Amazon SQS Queue" --execution-parameters "{\"VpcId\":
 \"VPC_ID\", \"StackId\": \"STACK_ID\", \"Parameters\": {\"SQSDelaySeconds\": 0,
 \"SQSMaximumMessageSize\": 262144, \"SQSMessageRetentionPeriod\": 345600,
 \"SQSQueueName\": \"MyQueueName\", \"SQSReceiveMessageWaitTimeSeconds\": 0,
 \"SQSVisibilityTimeout\": 60}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it UpdateSqsInstanceParams.json.

```
aws amscm get-change-type-version --change-type-id "ct-0hi7z7tyikjf6"
 --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
 UpdateSqsInstanceParams.json
```

2. Modify and save the UpdateSqsInstanceParams file. For example, you can replace the contents with something like this:

```
{
  "VpcId": "VPC_ID",
  "StackId": "STACK_ID",
  "Parameters": {
    "SQSDelaySeconds": 0,
    "SQSMaximumMessageSize": 262144,
    "SQSMessageRetentionPeriod": 345600,
    "SQSQueueName": "MyQueueName",
    "SQSReceiveMessageWaitTimeSeconds": 0,
    "SQSVisibilityTimeout": 60
  }
}
```

3. Output the RFC template JSON file to a file named UpdateSqsInstanceRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdateSqsInstanceRfc.json
```

4. Modify and save the UpdateSqsInstanceRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-0hi7z7tyikjf6",
"Title":                "Sqs-Instance-Update-RFC"
}
```

5. Create the RFC, specifying the UpdateSqsTopicSub Rfc file and the UpdateSqsTopicSubParams file:

```
aws amscm create-rfc --cli-input-json file://UpdateSqsInstanceRfc.json  --
execution-parameters file://UpdateSqsInstanceParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about Amazon Simple Queue Service (SQS), see Amazon Simple Queue Service.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0hi7z7tyikjf6.

Example: Required Parameters

```
{
  "VpcId": "vpc-01234567890abcdef",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "SQSQueueName": "mytestsqs"
  }
}
```

Example: All Parameters

```
{
  "VpcId": "vpc-12345678",
  "StackId": "stack-a1b2c3d4e5f67890e",
  "Parameters": {
    "SQSDelaySeconds": 0,
    "SQSMaximumMessageSize": 262144,
    "SQSMessageRetentionPeriod": 345600,
    "SQSQueueName": "mytestsqs",
    "SQSReceiveMessageWaitTimeSeconds": 0,
    "SQSVisibilityTimeout": 0
  }
}
```

Other Subcategory

Change Type Items and Operations in the Other Subcategory

Other | Create (Review Required)

Use to request manual creation of a resource.

Full classification: Management | Other | Other | Create (review required)

Change Type Details

Change type ID	ct-1e1xtak34nx76
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Create Other Other CTs

Creating an Other Other Create RFC with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Other Other Create RFC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1e1xtak34nx76" --change-type-version "1.0" --
title "TITLE" --execution-parameters "{\"Comment\": \"WHAT_TO_CREATE\"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named OtherCreateParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-1e1xtak34nx76" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > OtherCreateParams.json
   ```

2. Modify and save the OtherCreateParams file (example includes optional `Priority` parameter). For example, you can replace the contents with something like this:

   ```
   {
   "Comment":          "WHAT-TO-CREATE",
   "Priority":         "Medium"
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it OtherCreateRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > OtherCreateRfc.json
   ```

4. Modify and save the OtherCreateRfc.json file. For example, you can replace the contents with something like this:.

   ```
   {
   "ChangeTypeId":         "ct-1e1xtak34nx76",
   "ChangeTypeVersion":    "1.0",
   "Title":                "TITLE"
   }
   ```

5. Create the RFC, specifying the OtherCreateRfc file and the OtherCreateParams file:

```
aws amscm create-rfc --cli-input-json file://OtherCreateRfc.json  --execution-
parameters file://OtherCreateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> To update an existing resource, use Update Other Other CTs.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Use this CT when you can't find a change type for what you want; however, if you are unsure about specifying parameters in an existing CT, it is better to submit a service request for help. For information on submitting service requests, see Service Request Examples.

To update an existing resource, use Update Other Other CTs.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1e1xtak34nx76.

Example: Required Parameters

```
{
  "Comment": "This is a test comment"
}
```

Example: All Parameters

```
{
```

```
  "Comment": "This is a test comment",
  "Priority": "High",
  "RelatedIds": ["foo", "bar", "baz"]
}
```

Other | Update (Review Required)

Use to request a manual update to a resource.

Full classification: Management | Other | Other | Update (review required)

Change Type Details

Change type ID	ct-0xdawir96cy7k
Current version	1.0
Expected execution duration	240 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Update Other Other CTs

Creating an Other Other Update RFC with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Creating an Other Other Update RFC with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0xdawir96cy7k" --change-type-version "1.0"
 --title "TITLE" --execution-parameters "{\"Comment\": \"What you want changed\",
 \"Priority\": \"Medium\" \"RelatedIds\":[\"RESOURCE_ID\",\"RESOURCE_ID\"]}}"}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file named OtherUpdateParams.json.

   ```
   aws amscm get-change-type-version --change-type-id "ct-0xdawir96cy7k" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > OtherUpdateParams.json
   ```

2. Modify and save the OtherUpdateParams file. For example, you can replace the contents with something like this:

   ```
   {
   "Comment":          "WHAT-TO-UPDATE",
   "Priority":         "Medium",
   "RelatedIds":       ["RESOURCE_ID", "RESOURCE_ID"]
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it OtherUpdateRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > OtherUpdateRfc.json
   ```

4. Modify and save the OtherUpdateRfc.json file. For example, you can replace the contents with something like this:.

   ```
   {
   "ChangeTypeId":         "ct-0xdawir96cy7k",
   "ChangeTypeVersion":    "1.0",
   "Title":                "TITLE"
   }
   ```

5. Create the RFC, specifying the OtherUpdateRfc file and the OtherUpdateParams file:

```
aws amscm create-rfc --cli-input-json file://OtherUpdateRfc.json  --execution-
parameters file://OtherUpdateParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Important

Updating or Deleting stacks can have unwanted and unanticipated consequences. AMS prefers to *not* update or delete stacks or stack resources on behalf of customers for this reason. Note, that AMS will only update or delete resources on your behalf (through a submitted Mangement | Other | Other | Update change type) that are not possible to update or delete using the appropriate, automated, change type to delete.

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

Use this CT when you can't find a change type for what you want; however, if you are unsure about specifying parameters in an existing CT, it is better to submit a service request for help. For information on submitting service requests, see Service Request Examples.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0xdawir96cy7k.

Example: Required Parameters

```
{
  "Comment": "This is a test comment"
}
```

Example: All Parameters

```
{
  "Comment": "This is a test comment",
  "Priority": "High",
  "RelatedIds": ["foo", "bar", "baz"]
}
```

Patching Subcategory

Change Type Items and Operations in the Patching Subcategory

- On Demand Patching | Run
- Patch Window | Set Status
- Patch Window | Update

On Demand Patching | Run

Run on-demand SSM patching on specified instances; either a list of instances or instances with the specified tag/key pair.

Full classification: Management | Patching | On demand patching | Run

Change Type Details

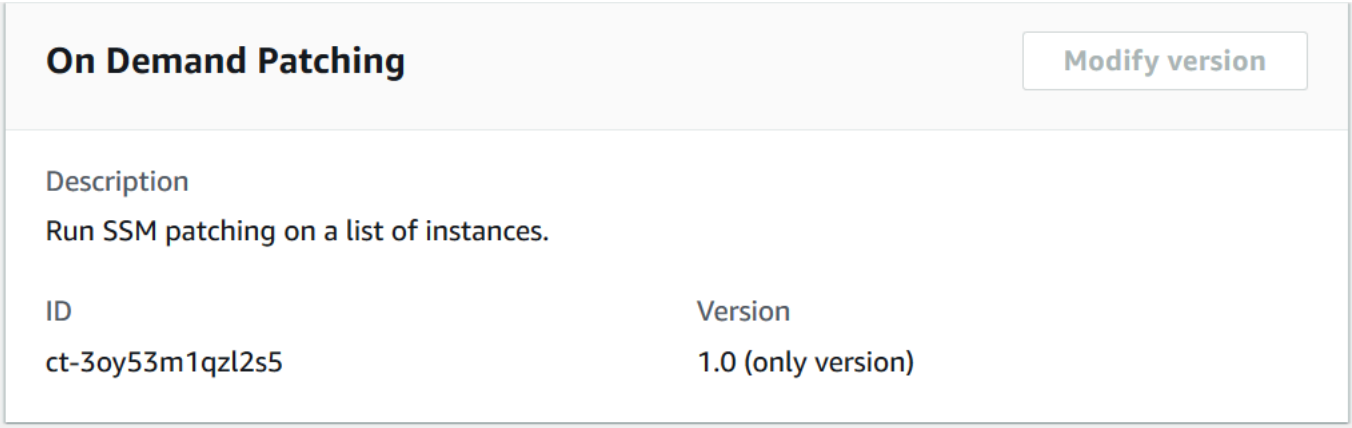
Change type ID	ct-3oy53m1qzl2s5
Current version	1.0
Expected execution duration	3600 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Run on-demand patching

Running Patching On Demand with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Running Patching On Demand with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
   aws amscm create-rfc --title my-test-patchbaseline --change-type-
id ct-3oy53m1qzl2s5 --change-type-version 1.0 --execution-parameters
 '{"Name":"test-ODP","PatchingTargets":[{"Key":"tag:Patch Group","Values":"[test-
odp]"}],"StartInactiveInstances":"True", "BackupVaultName":"test-backup-vault-name",
 "BackupIamRole":"test-backup-iam-role", "BackupRetentionInDays":"10"}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it RunOndemandPatchParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3oy53m1qzl2s5"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    RunOndemandPatchParams.json
   ```

2. Modify and save the RunOndemandPatchParams file.

   ```
   {
     "Name": "test-ODP",
     "PatchingTargets": [
       {
         "Key": "InstanceIds",
         "Values": ["INSTANCE_ID"]
       }
     ],
     "StartInactiveInstances": "True",
     "BackupVaultName": "test-backup-vault-name",
     "BackupIamRole": "test-backup-iam-role",
     "BackupRetentionInDays": "10"
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it RunOndemandPatchRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RunOndemandPatchRfc.json
   ```

4. Modify and save the RunOndemandPatchRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "1.0",
"ChangeTypeId":         "ct-3oy53m1qzl2s5",
"Title":                "Run-Ondemand-Patch-RFC"
}
```

5. Create the RFC, specifying the RunOndemandPatchRfc file and the RunOndemandPatchParams file:

```
aws amscm create-rfc --cli-input-json file://RunOndemandPatchRfc.json --execution-
parameters file://RunOndemandPatchParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Inactive instances**
>
> If you set **StartInactiveInstances** to **True**, inactive instances are started, patched, and returned to their original stopped state after patching is completed.

> **Concurrency limits**
>
> Any concurrently running maintenance windows will impact the safe limit execution of this RFC. This is because SSM automation is limited to 100 concurrent runs per account. Automations exceeding the concurrency limit are added to a queue of up to 1,000 executions. If a maintenance window is running on 50 instances, that leaves room for only 50 more to be concurrent runs for on-demand patching, adding any additional instances to the queue to be run once the in-progress executions are complete.

To learn more about AMS Patch Orchestrator and setting up SSM patch baseline and patch window, see Patch Orchestrator in the AMS Multi-account landing zone User Guide.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3oy53m1qzl2s5.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "Description": "description",
  "Name": "Test1",
  "PatchingTargets": [
    {
      "Key": "tag:Patch Group",
      "Values": [
        "testGroup"
      ]
    }
  ],
  "StartInactiveInstances": "True",
  "BackupVaultName": "backup-vault-name",
  "BackupIamRole": "backup-iam-role",
  "BackupRetentionInDays": "1"
}
```

Patch Window | Set Status

Enable or disable an existing AWS Systems Manager (SSM) patch window. If the window is enabled, any task associated with it runs on the next occurrence of the maintenance window. If disabled, any future occurrences of the window no longer run. Occurrences of the window that are already running continue to run until completion.

Full classification: Management | Patching | Patch window | Set status

Change Type Details

Change type ID	ct-3vfxkiudtovm9

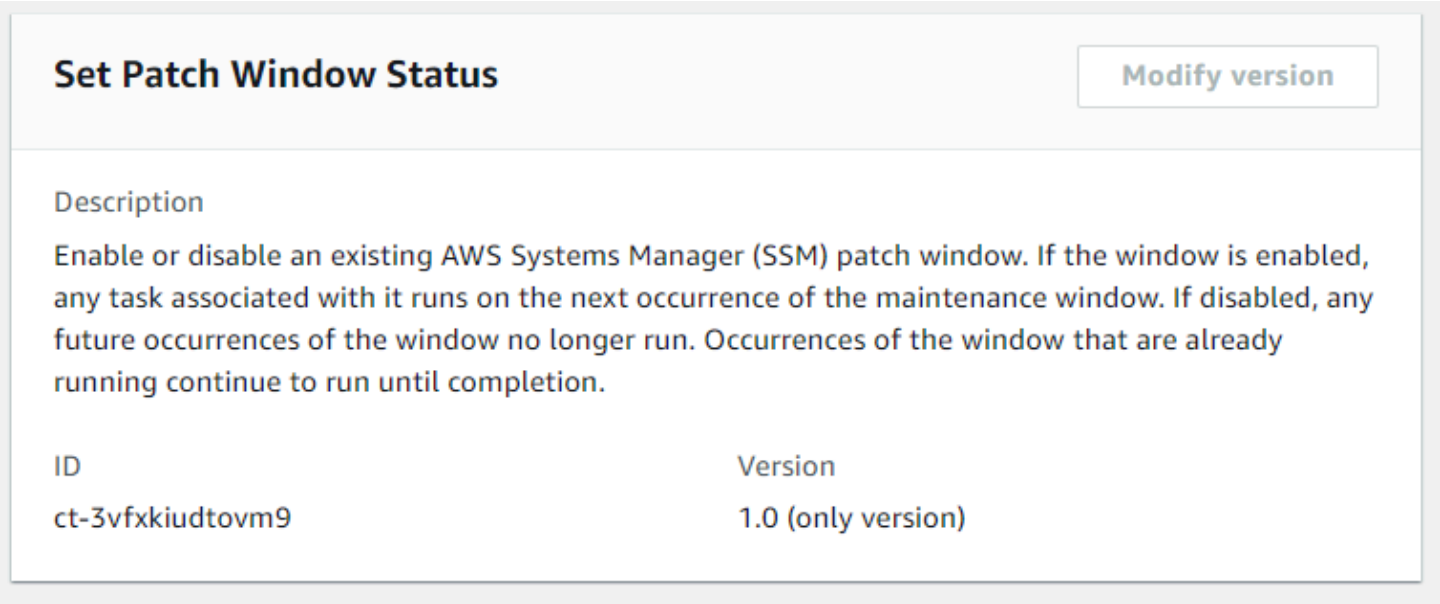
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Set SSM Patch window status

Set SSM Patch window status with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.

- **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

  To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Set SSM Patch window status with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3vfxkiudtovm9" --change-type-version
 "1.0" --title "Set patch window status" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-SetPatchenanceWindowStatus\",\"Region\":\"us-east-1\",\"Parameters
\":{\"MaintenanceWindowId\":[\"mw-1234567890abcdef0\"],\"Enabled\":[true]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it SetPatchWinStatusParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3vfxkiudtovm9"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    SetPatchWinStatusParams.json
   ```

2. Modify and save the SetPatchWinStatusParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-SetPatchenanceWindowStatus",
     "Region": "us-east-1",
     "Parameters":
   ```

```
  {
    "MaintenanceWindowId": [
      "mw-1234567890abcdef0"
    ],
    "Enabled": [
      true
    ]
  }
}
```

3. Output the RFC template to a file in your current folder; this example names it SetPatchWinStatusRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > SetPatchWinStatusRfc.json
```

4. Modify and save the SetPatchWinStatusRfc.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3vfxkiudtovm9",
  "Title": "Set patch window status"
}
```

5. Create the RFC, specifying the SetPatchWinStatusRfc file and the SetPatchWinStatusParams file:

```
aws amscm create-rfc --cli-input-json file://SetPatchWinStatusRfc.json  --
execution-parameters file://SetPatchWinStatusParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about doing this, see AWS Systems Manager Maintenance Windows.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3vfxkiudtovm9.

Example: Required Parameters

```
{
  "DocumentName" : "AWSManagedServices-SetSsmMaintenanceWindowStatus",
  "Region" : "us-east-1",
  "Parameters" : {
    "MaintenanceWindowId" : [
      "mw-1234567890abcdef0"
    ],
    "Enabled" : [
      true
    ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName" : "AWSManagedServices-SetSsmMaintenanceWindowStatus",
  "Region" : "us-east-1",
  "Parameters" : {
    "MaintenanceWindowId" : [
      "mw-1234567890abcdef0"
    ],
    "Enabled" : [
      true
    ]
  }
}
```

Patch Window | Update

Modify patch maintenance window settings created using version 1 of change type ct-0el2j07llrxs7.

Full classification: Management | Patching | Patch window | Update

Change Type Details

Change type ID	ct-2utx36abv83pv
Current version	2.0

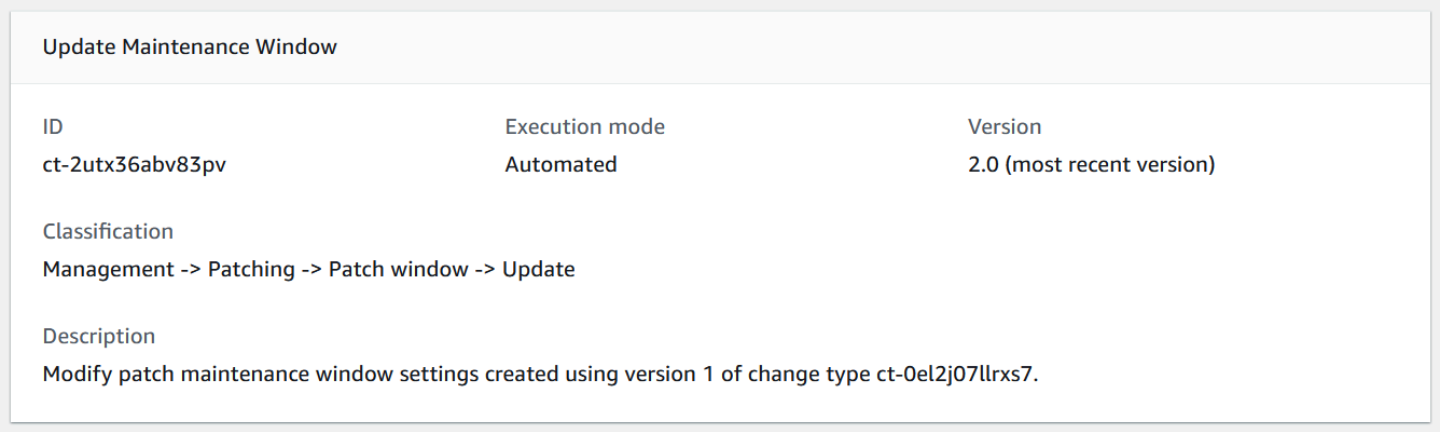
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update SSM Patch Window

Updating a patch window with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating a patch window with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotes when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-2utx36abv83pv" --change-type-version
 "2.0" --title "Patch-Window-Update-RFC" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-UpdateMaintenanceWindow\",\"Region\":\"us-east-1\",\"Parameters\":
{\"Duration\":[\"3\"],\"NotificationEmails\":[\"example@email.com\"],\"PatchGroupName
\":[\"MyApp\"],\"Schedule\":[\"cron(0 15 ? 9 TUE *)\"],\"ScheduleTimezone\":[\"UTC\"],
\"EmailAction\":[\"Add\"],\"OnlyCheckForMaintenanceWindowDrift\":[\"False\"],\"WindowId
\":[\"mw-012345678910abcef\"],\"BypassDriftDetection\":[\"False\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it UpdatePatchWindowParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2utx36abv83pv"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    UpdatePatchWindowParams.json
   ```

2. Modify and save the UpdatePatchWindowParams file.

   ```
   {
     "DocumentName": "AWSManagedServices-UpdateMaintenanceWindow",
     "Region": "us-east-1",
     "Parameters": {
       "Duration": [
         "3"
       ],
       "NotificationEmails": [
         "example@email.com"
       ],
   ```

```
    "PatchGroupName": [
      "MyApp"
    ],
    "Schedule": [
      "cron(0 15 ? 9 TUE *)"
    ],
    "ScheduleTimezone": [
      "UTC"
    ],
    "EmailAction": [
      "Add"
    ],
    "OnlyCheckForMaintenanceWindowDrift": [
      "False"
    ],
    "WindowId": [
      "mw-012345678910abcef"
    ],
    "BypassDriftDetection": [
        "False"
    ]
}
}
```

3. Output the RFC template to a file in your current folder; this example names it UpdatePatchWindowRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > UpdatePatchWindowRfc.json
```

4. Modify and save the UpdatePatchWindowRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion":    "2.0",
"ChangeTypeId":         "ct-2utx36abv83pv",
"Title":                "Patch-Window-Update-RFC"
}
```

5. Create the RFC, specifying the UpdatePatchWindowRfc file and the UpdatePatchWindowParams file:

```
aws amscm create-rfc --cli-input-json file://UpdatePatchWindowRfc.json --execution-
parameters file://UpdatePatchWindowParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

6. To view the SSM patch baseline, look in the execution output: Use the stack_id to view the patch baseline in the Systems Manager console.

Tips

- This solution uses custom logic to detect drift in AWS resources not yet supported by CloudFormation (AWS::SSM::MaintenanceWindow & AWS::SSM::MaintenanceWindowTarget). We recommend executing this change type with the parameter **OnlyCheckForMaintenanceWindowDrift=True** for the most accurate report of drifted AMS Patch maintance window resources.
- To learn more about AWS SSM patch windows, see "Maintenance Window" at Patching your Windows EC2 instances using AWS Systems Manager Patch Manager.
- To delete a custom Maintenance Window, see Delete stack.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2utx36abv83pv.

Example: Required Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateMaintenanceWindow",
  "Region": "us-east-1",
  "Parameters": {
    "EmailAction": [ "None" ],
    "OnlyCheckForMaintenanceWindowDrift": [ "True" ],
    "WindowId": [ "mw-012345678910abcef" ],
    "BypassDriftDetection": [ "False" ]
  }
}
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-UpdateMaintenanceWindow",
  "Region": "us-east-1",
  "Parameters": {
    "Duration": [ "2" ],
    "EmailAction": [ "Add" ],
    "NotificationEmails": [ "nobody@amazon.com" ],
    "OnlyCheckForMaintenanceWindowDrift": [ "False" ],
    "PatchGroupName": [ "Test" ],
    "Schedule": [ "cron(0 17 * * ? *)" ],
    "ScheduleTimezone": [ "UTC" ],
    "WindowId": [ "mw-012345678910abcef" ],
    "BypassDriftDetection": [ "False" ]
  }
}
```

Standalone Resources Subcategory

Change Type Items and Operations in the Standalone Resources Subcategory

- EC2 Instance | Terminate
- RDS Instance | Terminate

EC2 Instance | Terminate

Terminate up to fifty EC2 instances. The automation checks that none of the instances are part of an Auto Scaling group and none have termination protection enabled. Instances meeting either of those criteria are not terminated. Standalone resources for testing purposes are created by AMS upon your request, they are not part of a stack and can't be deleted with ct-0q0bic0ywqk6c.

Full classification: Management | Standalone resources | EC2 instance | Terminate

Change Type Details

Change type ID	ct-3dfubbpesm2v9
Current version	1.0
Expected execution duration	150 minutes

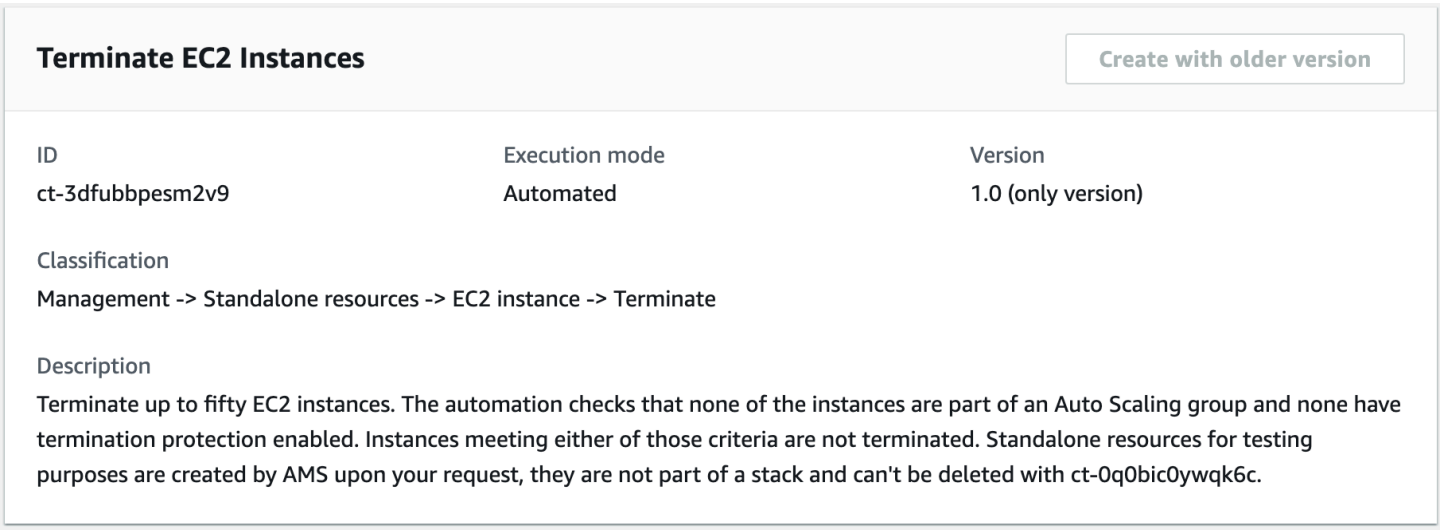
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Terminate instances

Terminating standalone EC2 instances with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Terminating standalone EC2 instances with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status

changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Only specify the parameters you want to change. Absent parameters retain the existing values.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3dfubbpesm2v9" --change-type-version "1.0"
 --title "Terminate standalone instances" --execution-parameters "{\"DocumentName
\": \"AWSManagedServices-TerminateStandaloneInstances\",\"Region\": \"us-east-1\",
\"Confirmation\": \"terminate instances\",\"Parameters\": {\"InstanceIds\":
 [\"i-1234567890abcdef0\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it TerminateStandaloneEc2sParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3dfubbpesm2v9"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TerminateStandaloneEc2sParams.json
   ```

2. Modify and save the TerminateStandaloneEc2sParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-TerminateStandaloneInstances",
     "Region": "us-east-1",
     "Confirmation": "terminate instances",
     "Parameters": {
       "InstanceIds": [
         "i-1234567890abcdef0"
       ]
     }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it TerminateStandaloneEc2sRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > TerminateStandaloneEc2sRfc.json
```

4. Modify and save the TerminateStandaloneEc2sRfc.json file. For example, you can replace the contents with something like this:

```
{
    "ChangeTypeId": "ct-3dfubbpesm2v9",
    "ChangeTypeVersion": "1.0",
    "Title": "Terminate standalone EC2 instance"
}
```

5. Create the RFC, specifying the TerminateStandaloneEc2sRfc file and the TerminateStandaloneEc2sParams file:

```
aws amscm create-rfc --cli-input-json file://TerminateStandaloneEc2sRfc.json  --
execution-parameters file://TerminateStandaloneEc2sParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about termination protection, see How do I protect my data against accidental EC2 instance termination?.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3dfubbpesm2v9.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-TerminateStandaloneInstances",
    "Region": "us-east-1",
    "Confirmation": "terminate instances",
```

```
    "Parameters": {
        "InstanceIds": [
            "i-1234567890abcdef0"
        ]
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-TerminateStandaloneInstances",
    "Region": "us-east-1",
    "Confirmation": "terminate instances",
    "Parameters": {
        "InstanceIds": ["i-1234567890abcdef0"]
    }
}
```

RDS Instance | Terminate

Terminate a standalone DB instance or cluster. The automation checks that the DB instance, or cluster, is not part of a CloudFormation stack and does not have termination protection enabled. Please note that deleting the DB cluster deletes all the automated backups for that DB cluster and those backups can't be recovered. Standalone resources for testing purposes are created by AMS upon your request, they are not part of a stack and they can't be deleted with ct-0q0bic0ywqk6c.

Full classification: Management | Standalone resources | RDS instance | Terminate

Change Type Details

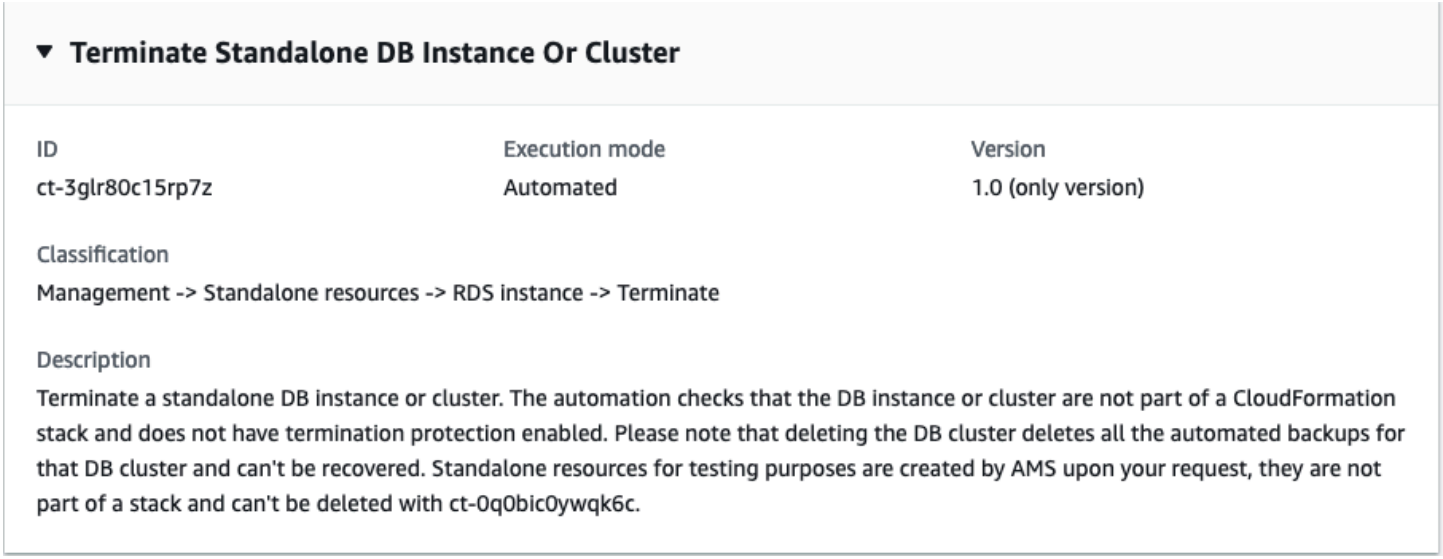
Change type ID	ct-3glr80c15rp7z
Current version	1.0
Expected execution duration	150 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Terminate Standalone RDS instance or cluster

Terminating standalone RDS instance or cluster with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Terminating standalone RDS instance or cluster with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Only specify the parameters you want to change. Absent parameters retain the existing values.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3glr80c15rp7z" --change-type-version
 "1.0" --title "Terminate Standalone DB Instance Or Cluster" --execution-
parameters "{\"DocumentName\": \"TerminateStandaloneDBInstanceOrCluster\",
\"Region\": \"us-east-1\",\"Parameters\": {\"DBIdentifierArn\": [\"arn:aws:rds:us-
east-1:123456789101:db:testdb-instance-1\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it TerminateStandaloneDBParameters.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3glr80c15rp7z"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    TerminateStandaloneDBParameters.json
   ```

2. Modify and save the TerminateStandaloneDBParameters.json file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   TerminateStandaloneDBParameters.json
   {
     "DocumentName": "TerminateStandaloneDBInstanceOrCluster",
     "Region": "us-east-1",
     "DBIdentifierArn": [
         "arn:aws:rds:us-east-1:123456789101:db:testdb-instance-1"
       ]
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it TerminateStandaloneDB.json:

```
aws amscm create-rfc --generate-cli-skeleton > TerminateStandaloneDB.json
```

4. Modify and save the TerminateStandaloneDB.json file. For example, you can replace the contents with something like this:

```
{
  "ChangeTypeVersion": "1.0",
  "ChangeTypeId": "ct-3glr80c15rp7z",
  "Title": "Terminate Standalone DB Instance Or Cluster"
}
```

5. Create the RFC, specifying the TerminateStandaloneDB.json file and the TerminateStandaloneDBParameters file:

```
aws amscm create-rfc --cli-input-json file://TerminateStandaloneDB.json --
execution-parameters file://TerminateStandaloneDBParameters.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

To learn more about RDS deletion protection, see Deletion Protection.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3glr80c15rp7z.

Example: Required Parameters

```
{
    "DocumentName": "AWSManagedServices-TerminateStandaloneDBInstanceOrCluster",
    "Region": "us-east-1",
    "Confirmation": "permanently delete",
    "Parameters": {
        "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance"
```

```
    }
}
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-TerminateStandaloneDBInstanceOrCluster",
    "Region": "us-east-1",
    "Confirmation": "permanently delete",
    "Parameters": {
      "DBIdentifierArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
      "CreateFinalSnapshot": true,
      "DeleteAutomatedBackups": false,
      "FinalDBSnapshotIdentifier": "final-db-snapshot"
    }
  }
```

Standard Stacks Subcategory

Change Type Items and Operations in the Standard Stacks Subcategory

- Stack | Delete
- Stack | Reboot
- Stack | Remediate Drift
- Stack | Remediate Drift (Review Required)
- Stack | Start
- Stack | Stop
- Stack | Update Termination Protection

Stack | Delete

Delete an existing stack and its resources from your account. The effects of deleting a resource vary. For details, see the appropriate AWS documentation for the resource. Note that termination protection on a resource in the stack causes the RFC to fail. To check for a resource's termination protection status, see the corresponding AWS console.

Full classification: Management | Standard stacks | Stack | Delete

Change Type Details

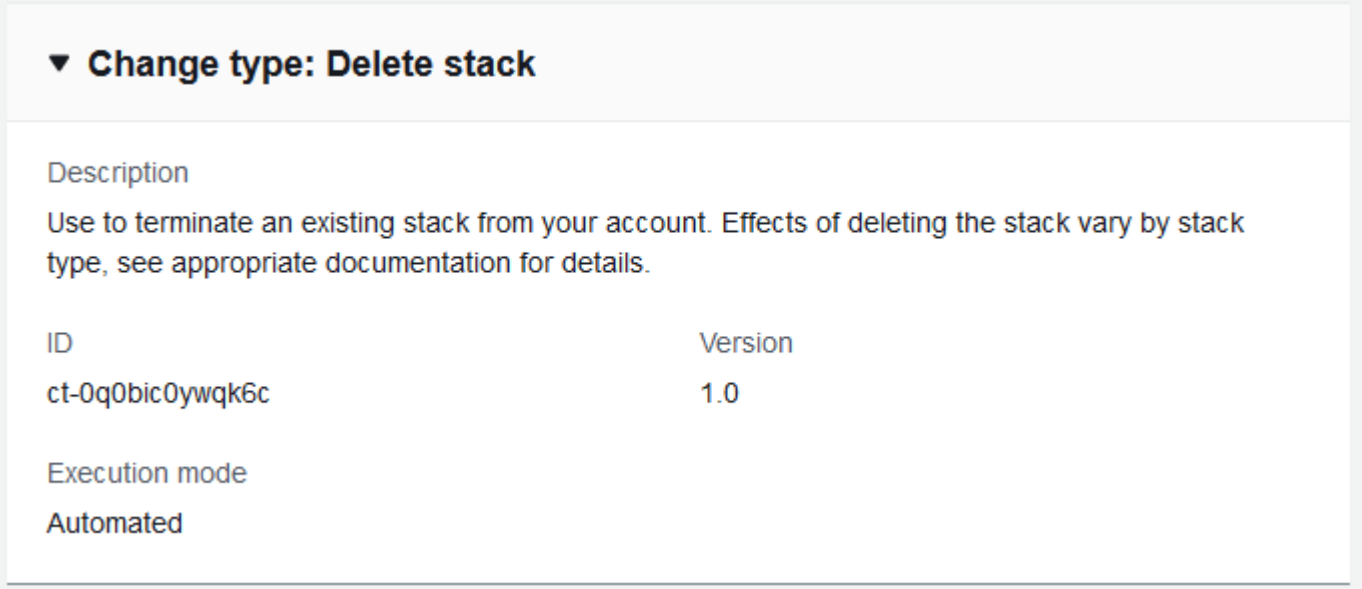
Change type ID	ct-0q0bic0ywqk6c
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Delete stack

Deleting a Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Deleting a Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

Note

You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-0q0bic0ywqk6c" --change-type-version "1.0" --
title "Delete My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder; this example names it DeleteStackRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > DeleteStackRfc.json
   ```

2. Modify and save the DeleteStackRfc.json file.

   The internal quotation marks in the ExecutionParameters JSON extension must be escaped with a backslash (\). Example without start and end time:

   ```
   {
   "ChangeTypeVersion":    "1.0",
   "ChangeTypeId":         "ct-0q0bic0ywqk6c",
   "Title":                "Delete-My-Stack-RFC"
   ```

```
"ExecutionParameters":  "{
        \"StackId\":\"STACK_ID\"}"
}
```

3. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://DeleteStackRfc.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> If deleting an S3 bucket, it must be emptied of objects first.

> **Important**
>
> Deleting stacks can have unwanted and unanticipated consequences. For important caveats, see RFC Troubleshooting section RFCs for Delete Stack.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-0q0bic0ywqk6c.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "StackId": "stack-a1b2c3d4e5f67890e",
  "TimeoutInMinutes": 720
}
```

Stack | Reboot

Use to reboot all running EC2 and RDS DB instances in the specified stack.

Full classification: Management | Standard stacks | Stack | Reboot

Change Type Details

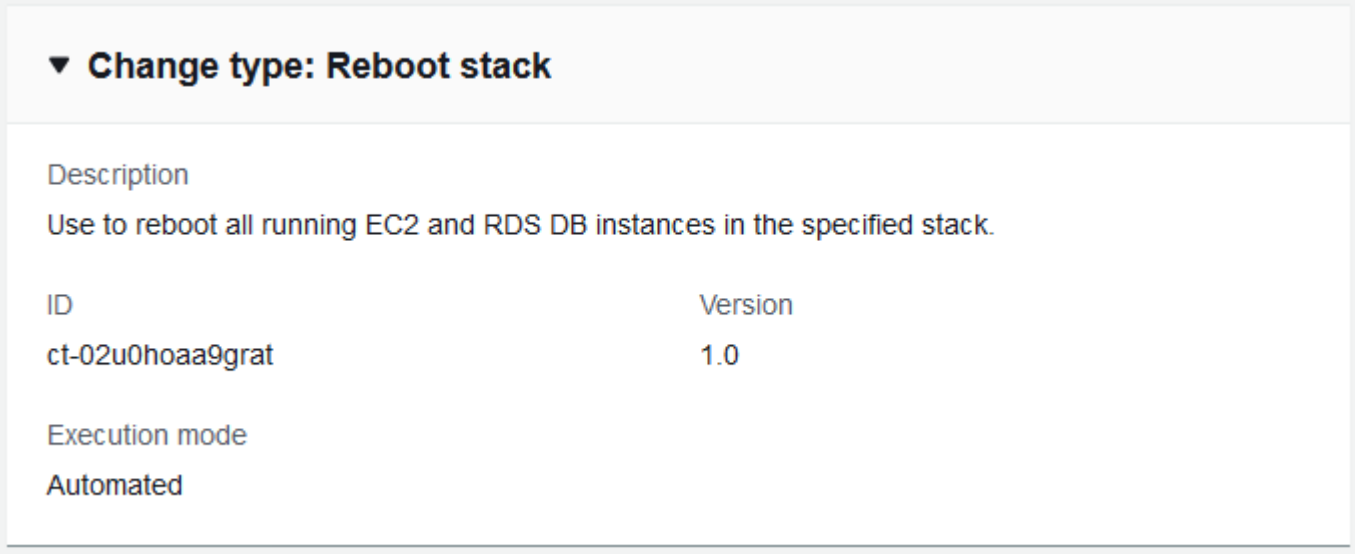
Change type ID	ct-02u0hoaa9grat
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Reboot stack

Rebooting a Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Rebooting a Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-02u0hoaa9grat" --change-type-version "1.0" --
title "Reboot My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder. This example names it RebootStackRfc.json. Note that since there is only one execution parameter for stopping (rebooting, or starting) an instance, the execution parameter can be in the schema JSON file itself and there is no need to create a separate execution parameters JSON file.

   ```
   aws amscm create-rfc --generate-cli-skeleton > StopInstanceRfc.json
   ```

2. Modify and save the RebootStackRfc.json file.

The internal quotation marks in the `ExecutionParameters` JSON extension must be escaped with a backslash (\). Example:

```
{
"ChangeTypeId":          "ct-02u0hoaa9grat",
"Title":                 "Reboot-My-EC2-RFC",
"TimeoutInMinutes":      60,
"ExecutionParameters":   "{
        \"StackId\":\"STACK_ID\"
    }"
}
```

3. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://RebootStackRfc.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Application Load Balancers, see Application Load Balancers.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-02u0hoaa9grat.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "StackId": "stack-f16bbbeea61df041f"
}
```

Stack | Remediate Drift

Remediate the drift (out-of-band changes) in a stack, bringing the stack in sync and enabling you to perform future updates using the available Update CTs. Note: up to 10 drifted resources will be remediated per RFC.

Full classification: Management | Standard stacks | Stack | Remediate drift

Change Type Details

Change type ID	ct-3kinq0u4l33zf
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Remediate stack drift

Remediating Stack Drift with the Console

Screenshot of this change type in the AMS console:

How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Remediating Stack Drift with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.

2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3kinq0u4l33zf" --change-type-version "1.0" --
title "Remediate Stack Drift, no ops review" --execution-parameters "{\"DocumentName\":
 \"AWSManagedServices-StartDriftRemediation\", \"Region\": \"us-east-1\", \"Parameters
\": {\"StackName\": [\"stack-xxxxxxxxxxxxxxxxx\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemediateDriftNrrParams.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftNrrParams.json
   ```

2. Modify and save the RemediateDriftNrrParams file. For example, you can replace the contents with something like this:

```
{
    "DocumentName": "AWSManagedServices-StartDriftRemediation",
    "Region": "us-east-1",
    "Parameters": {
        "StackName": [
            "stack-xxxxxxxxxxxxxxxxx"
        ]
    }
}
```

3. Output the RFC template JSON file to a file; this example names it RemediateDriftNrrRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RemediateDriftNrrRfc.json
```

4. Modify and save the RemediateDriftNrrRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId": "ct-3kinq0u4l33zf",
"ChangeTypeVersion": "1.0",
"Title": "Remediate stack drift, no ops review"
}
```

5. Create the RFC, specifying the RemediateDriftNrrRfc file and the RemediateDriftNrrParams file:

```
aws amscm create-rfc --cli-input-json file://RemediateDriftNrrRfc.json  --
execution-parameters file://RemediateDriftNrrParams.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

Important

Stack remediation modifies the stack template and/or parameter values. Once remediation is complete, you must update your local template repositories, or any automation, that would be updating the remediated stack, with the latest template and parameters provided

in the RFC summary of the remediation. It is very important to do this, because using the old template and/or parameters can cause destructive changes on the stack resources. For more details, including a list of *Limitations*, see Drift remediation FAQs.

Note

When using "review required" CTs, AMS recommends that you use the ASAP **Scheduling** option (choose **ASAP** in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run. If you schedule these RFCs, be sure to allow at least 24 hours. If approval does not happen before the scheduled start time, the RFC is rejected automatically.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3kinq0u4l33zf.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
    "DocumentName": "AWSManagedServices-StartDriftRemediation",
    "Region": "us-east-1",
    "Parameters": {
      "StackName": ["stack-a1b2c3d4e5f678900"],
      "DryRun": ["true"]
    }
}
```

Stack | Remediate Drift (Review Required)

Remediate the drift (out-of-band changes) in a stack, bringing the stack in sync and enabling you to perform future updates using the available Update CTs. Drift remediation can be performed on EC2 resource types.

Full classification: Management | Standard stacks | Stack | Remediate drift (review required)

Change Type Details

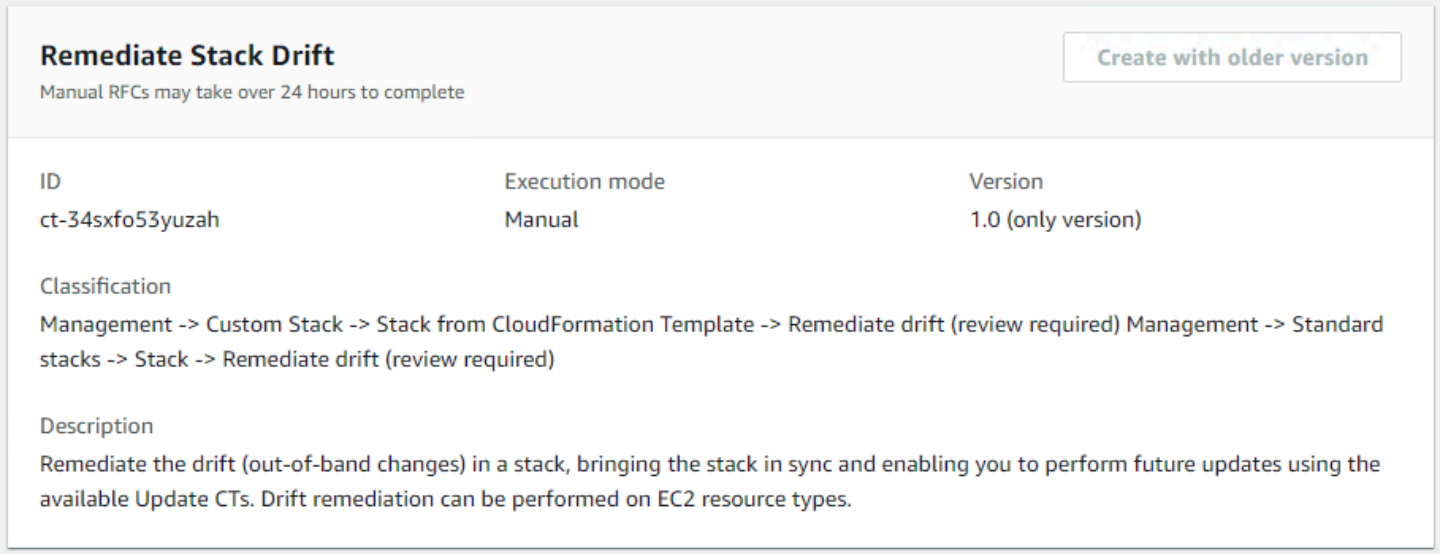
Change type ID	ct-34sxfo53yuzah
Current version	1.0
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required if submitter
Execution mode	Manual

Additional Information

Remediate stack drift (review required)

Remediating Stack Drift (review required) with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.

2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.
5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Remediating Stack Drift (review required) with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-34sxfo53yuzah" --change-type-version
 "1.0" --title "Remediate stack drift" --execution-parameters '{"StackName":"stack-
a1b2c3d4e5f67890e","DryRun":false}'
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a file; this example names it RemediateDriftParams.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > RemediateDriftParams.json
   ```

2. Modify and save the RemediateDriftParams file. For example, you can replace the contents with something like this:

   ```
   {
   "StackName" : "stack-a1b2c3d4e5f67890e",
   "DryRun" : false
   }
   ```

3. Output the RFC template JSON file to a file; this example names it RemediateDriftRfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > RemediateDriftRfc.json
```

4. Modify and save the RemediateDriftRfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeId": "ct-34sxfo53yuzah",
"ChangeTypeVersion": "1.0",
"Title": "Remediate stack drift"
}
```

5. Create the RFC, specifying the RemediateDriftRfc file and the RemediateDriftParams file:

```
aws amscm create-rfc --cli-input-json file://RemediateDriftRfc.json  --execution-
parameters file://RemediateDriftParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

This is a "review required" change type (an AMS operator must review and run the CT), which means that the RFC can take longer to run and you might have to communicate with AMS through the RFC details page correspondance option. Additionally, if you schedule a "review required" change type RFC, be sure to allow at least 24 hours, if approval does not happen before the scheduled start time, the RFC is rejected automatically.

> **Note**
>
> When using "review required" CTs, AMS recommends that you use the ASAP **Scheduling** option (choose **ASAP** in the console, leave start and end time blank in the API/CLI) as these CTs require an AMS operator to examine the RFC, and possibly communicate with you before it can be approved and run. If you schedule these RFCs, be sure to allow at least 24 hours. If approval does not happen before the scheduled start time, the RFC is rejected automatically.

- There is an automated version of this change type that runs more quickly, though there are some limitations. For more details, see Stack | Remediate Drift.
- Stack remediation modifies the stack template and/or parameter values. Once remediation is complete, you must update your local template repositories, or any automation, that would be updating the remediated stack, with the latest template and parameters provided in the RFC summary of the remeditation. It is very important to do this, because using the old template and/or parameters can cause destructive changes on the stack resources.

  For more details, see Drift remediation FAQs.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-34sxfo53yuzah.

Example: Required Parameters

```
{
  "StackName": "stack-a1b2c3d4e5f678900"
}
```

Example: All Parameters

```
{
  "StackName": "stack-a1b2c3d4e5f678900",
  "DryRun": false,
  "Priority": "Medium"
}
```

Stack | Start

Use to start all stopped EC2 instances in the specified stack.

Full classification: Management | Standard stacks | Stack | Start

Change Type Details

Change type ID	ct-1h5xgl9cr4bzy
Current version	1.0

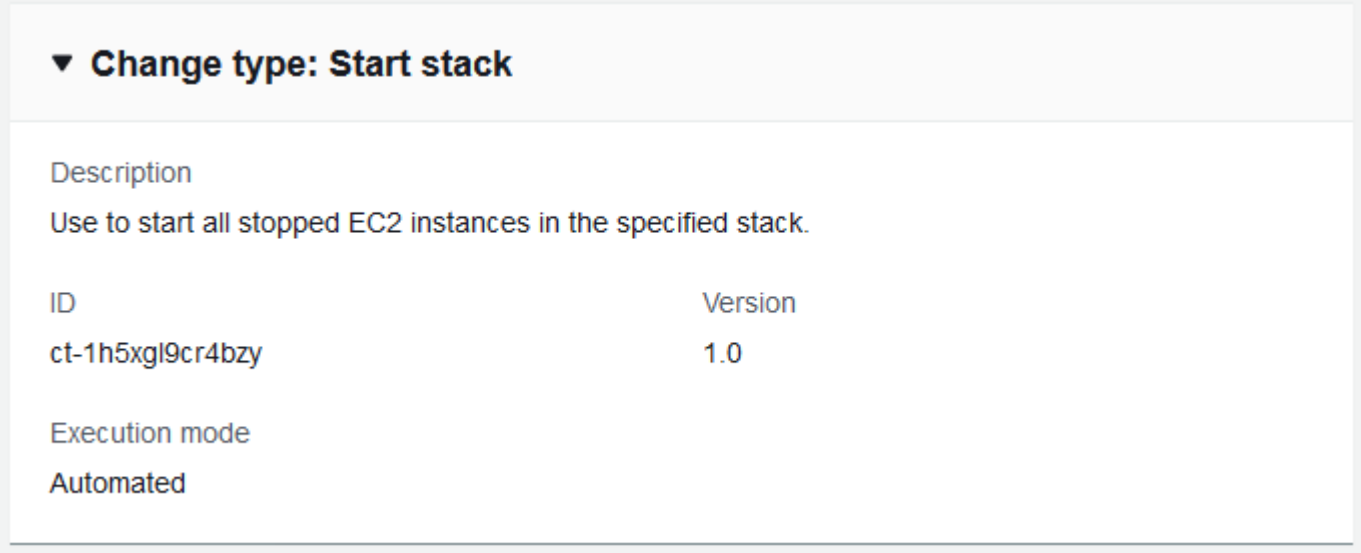
Expected execution duration	60 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Start stack

Starting a Stack with the Console

Screenshot of this change type in the AMS console:



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.

- **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Starting a Stack with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-1h5xgl9cr4bzy" --change-type-version "1.0" --
title "Start My Stack" --execution-parameters "{\"StackId\":\"STACK_ID\"}"
```

TEMPLATE CREATE:

1. Output the RFC template to a file in your current folder. This example names it StartInstanceRfc.json. Note that since there is only one execution parameter for starting a stack, the execution parameter can be in the schema JSON file itself and there is no need to create a separate execution parameters JSON file.

   ```
   aws amscm create-rfc --generate-cli-skeleton > StartStackRfc.json
   ```

2. Modify and save the StartStackRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
   "ChangeTypeId":          "ct-1h5xgl9cr4bzy",
   "Title":                 "Start-My-EC2-RFC",
   "TimeoutInMinutes":      60,
   "ExecutionParameters":   "{
           \"StackId\":\"STACK_ID\"
       }"
   }
   ```

3. Create the RFC:

```
aws amscm create-rfc --cli-input-json file://StartStackRfc.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

For information about Application Load Balancers, see Application Load Balancers.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-1h5xgl9cr4bzy.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "StackId": "stack-f16bbbeea61df041f"
}
```

Stack | Stop

Use to stop all running EC2 instances in the specified stack.

Full classification: Management | Standard stacks | Stack | Stop

Change Type Details

Change type ID	ct-3dgbnh6gpst4d
Current version	1.0
Expected execution duration	60 minutes

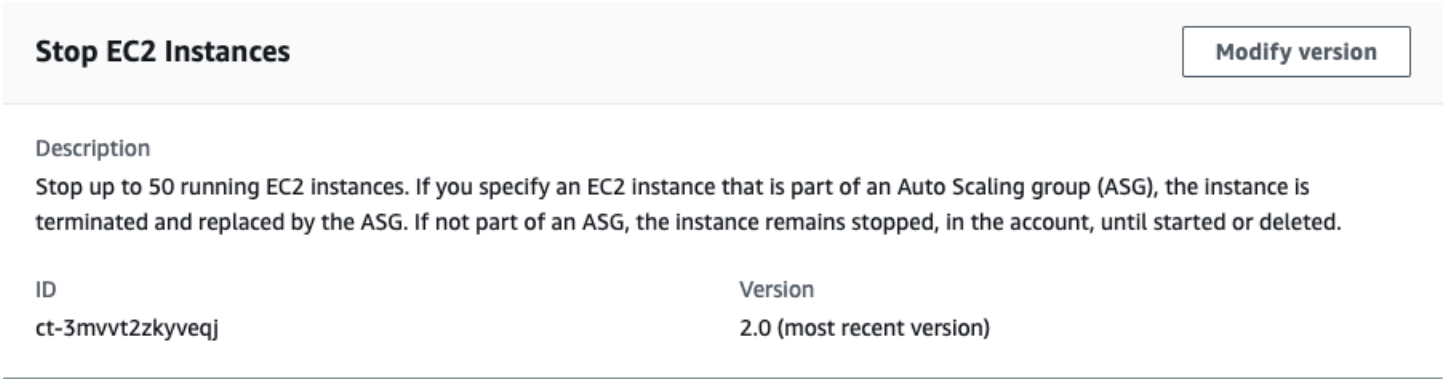
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Stop stack

Stopping an EC2 instance with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.

3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.

4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Stopping an EC2 instance with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients \" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not

the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc --change-type-id "ct-3mvvt2zkyveqj" --change-type-version
 "3.0" --title "Stop EC2 Instances" --execution-parameters "{\"DocumentName\":
\"AWSManagedServices-StopInstances\",\"Region\":\"us-east-1\",\"Parameters\":
{\"InstanceIds\":[\"i-1234567890abcdef0\", \"i-1234567890abcdef1\"],\"ForceStop\":
[\"false\"],\"StopASGInServiceInstances\":[\"false\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters JSON schema for this change type to a JSON file; this example names it StopEC2Params.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-3mvvt2zkyveqj" --query
    "ChangeTypeVersion.ExecutionInputSchema" --output text > StopEC2Params.json
   ```

2. Modify and save the StopEC2Params file.

   ```
   {
   "DocumentName" : "AWSManagedServices-StopInstances",
   "Region" : "us-east-1",
   "Parameters" : {
       "InstanceIds" : [
       "i-1234567890abcdef0",
       "i-1234567890abcdef1"
       ],
       "ForceStop": [
       "false"
       ],
       "StopASGInServiceInstances": [
       "false"
       ]
   }
   ```

```
}
```

3. Output the RFC template to a file in your current folder; this example names it StopEC2Rfc.json:

```
aws amscm create-rfc --generate-cli-skeleton > StopEC2Rfc.json
```

4. Modify and save the StopEC2Rfc.json file. For example, you can replace the contents with something like this:

```
{
"ChangeTypeVersion": "3.0",
"ChangeTypeId": "ct-3mvvt2zkyveqj",
"Title": "Stop EC2 Instances"
}
```

5. Create the RFC, specifying the StopEC2Rfc file and the StopEC2Params file:

```
aws amscm create-rfc --cli-input-json file://StopEC2Rfc.json --execution-parameters
 file://StopEC2Params.json
```

You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> This change type is now at version 3.0. The schema has been changed so you can stop up to fifty instances.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-3dgbnh6gpst4d.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "StackId": "stack-f16bbbeea61df041f"
}
```

Stack | Update Termination Protection

Update existing defined termination protection for stacks.

Full classification: Management | Standard stacks | Stack | Update termination protection

Change Type Details

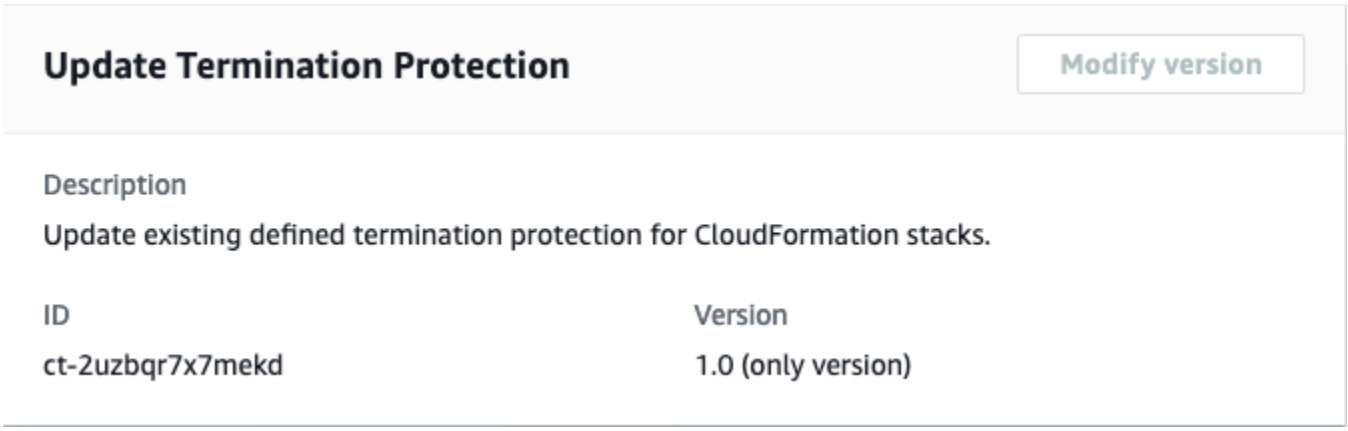
Change type ID	ct-2uzbqr7x7mekd
Current version	1.0
Expected execution duration	10 minutes
AWS approval	Required
Customer approval	Not required
Execution mode	Automated

Additional Information

Update AWS CloudFormation stacks termination protection

Updating an AWS CloudFormation termination protection stack with the console

The following shows this change type in the AMS console.



How it works:

1. Navigate to the **Create RFC** page: In the left navigation pane of the AMS console click **RFCs** to open the RFCs list page, and then click **Create RFC**.
2. Choose a popular change type (CT) in the default **Browse change types** view, or select a CT in the **Choose by category** view.
   - **Browse by change type**: You can click on a popular CT in the **Quick create** area to immediately open the **Run RFC** page. Note that you cannot choose an older CT version with quick create.

     To sort CTs, use the **All change types** area in either the **Card** or **Table** view. In either view, select a CT and then click **Create RFC** to open the **Run RFC** page. If applicable, a **Create with older version** option appears next to the **Create RFC** button.
   - **Choose by category**: Select a category, subcategory, item, and operation and the CT details box opens with an option to **Create with older version** if applicable. Click **Create RFC** to open the **Run RFC** page.
3. On the **Run RFC** page, open the CT name area to see the CT details box. A **Subject** is required (this is filled in for you if you choose your CT in the **Browse change types** view). Open the **Additional configuration** area to add information about the RFC.

   In the **Execution configuration** area, use available drop-down lists or enter values for the required parameters. To configure optional execution parameters, open the **Additional configuration** area.
4. When finished, click **Run**. If there are no errors, the **RFC successfully created** page displays with the submitted RFC details, and the initial **Run output**.

5. Open the **Run parameters** area to see the configurations you submitted. Refresh the page to update the RFC execution status. Optionally, cancel the RFC or create a copy of it with the options at the top of the page.

Updating an AWS CloudFormation stack termination protection with the CLI

How it works:

1. Use either the Inline Create (you issue a `create-rfc` command with all RFC and execution parameters included), or Template Create (you create two JSON files, one for the RFC parameters and one for the execution parameters) and issue the `create-rfc` command with the two files as input. Both methods are described here.
2. Submit the RFC: `aws amscm submit-rfc --rfc-id` *`ID`* command with the returned RFC ID.

   Monitor the RFC: `aws amscm get-rfc --rfc-id` *`ID`* command.

To check the change type version, use this command:

```
aws amscm list-change-type-version-summaries --filter
 Attribute=ChangeTypeId,Value=CT_ID
```

> **Note**
>
> You can use any `CreateRfc` parameters with any RFC whether or not they are part of the schema for the change type. For example, to get notifications when the RFC status changes, add this line, `--notification "{\"Email\": {\"EmailRecipients\" : [\"email@example.com\"]}}"` to the RFC parameters part of the request (not the execution parameters). For a list of all CreateRfc parameters, see the AMS Change Management API Reference.

Only specify the parameters you want to change. Absent parameters retain the existing values.

INLINE CREATE:

Issue the create RFC command with execution parameters provided inline (escape quotation marks when providing execution parameters inline), and then submit the returned RFC ID. For example, you can replace the contents with something like this:

```
aws amscm create-rfc \
--change-type-id "ct-2uzbqr7x7mekd" \
--change-type-version "1.0" \
--title "Enable termination protection on CFN stack" \
--execution-parameters "{\"DocumentName\":\"AWSManagedServices-
ManageResourceTerminationProtection\",\"Region\":\"us-east-1\",\"Parameters\":
{\"ResourceId\":[\"stack-psvnq6cupymio3enl\"],\"TerminationProtectionDesiredState\":
[\"enabled\"]}}"
```

TEMPLATE CREATE:

1. Output the execution parameters for this change type to a JSON file; this example names it EnableTermProCFNParams.json:

   ```
   aws amscm get-change-type-version --change-type-id "ct-2uzbqr7x7mekd"
    --query "ChangeTypeVersion.ExecutionInputSchema" --output text >
    EnableTermProCFNParams.json
   ```

2. Modify and save the EnableTermProCFNParams file, retaining only the parameters that you want to change. For example, you can replace the contents with something like this:

   ```
   {
     "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
     "Region": "us-east-1",
     "Parameters": {
       "ResourceId": ["stack-psvnq6cupymio3enl"],
       "TerminationProtectionDesiredState": ["enabled"]
     }
   }
   ```

3. Output the RFC template to a file in your current folder; this example names it EnableTermProCFNRfc.json:

   ```
   aws amscm create-rfc --generate-cli-skeleton > EnableTermProCFNRfc.json
   ```

4. Modify and save the EnableTermProCFNRfc.json file. For example, you can replace the contents with something like this:

   ```
   {
       "ChangeTypeId": "ct-2uzbqr7x7mekd",
       "ChangeTypeVersion": "1.0",
   ```

```
    "Title": "Enable termination protection on CFN instance"
}
```

5. Create the RFC, specifying the EnableTermProCFNRfc file and the EnableTermProCFNParams file:

```
aws amscm create-rfc --cli-input-json file://EnableTermProCFNRfc.json  --execution-
parameters file://EnableTermProCFNParams.json
```

   You receive the ID of the new RFC in the response and can use it to submit and monitor the RFC. Until you submit it, the RFC remains in the editing state and does not start.

Tips

> **Note**
>
> There is a related CT for Amazon EC2, EC2 stack: Updating termination protection.

To learn more about termination protection, see Protecting a stack from being deleted.

Execution Input Parameters

For detailed information about the execution input parameters, see Schema for Change Type ct-2uzbqr7x7mekd.

Example: Required Parameters

```
Example not available.
```

Example: All Parameters

```
{
  "DocumentName": "AWSManagedServices-ManageResourceTerminationProtection",
  "Region": "eu-west-1",
  "Parameters": {
    "ResourceId": ["stack-1234567890abcd"],
    "TerminationProtectionDesiredState": ["enabled"]
  }
}
```

Change Type Schemas

Change type schemas specify the execution input parameters for a change type.

Schema for Change Type ct-00tlkda4242x7

Classifications:

- Deployment | Applications | CodeDeploy deployment group | Create (for EC2 instance)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create CodeDeploy deployment group for EC2 instance as target.",
  "description": "Create an AWS CodeDeploy application deployment group specifically
 for an EC2 instance as target. Tags you create in the EC2 instances, and specify here
 (EC2FilterTag1, 2, and 3), mark the instances as targets for the deployment group. A
 name for the deployment group is automatically generated.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
```

```
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-n3hsoirgqeqqdbpk2",
      "type": "string",
      "enum": [
        "stm-n3hsoirgqeqqdbpk2"
      ],
      "default": "stm-n3hsoirgqeqqdbpk2"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
  the change. This will not prolong execution, but the RFC fails if the change is not
  completed in the specified time.",
```

```
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ApplicationName": {
          "type": "string",
          "description": "The name of an existing AWS CodeDeploy application within
your AMS account.",
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        },
        "DeploymentConfigName": {
          "type": "string",
          "description": "The configuration for deployment operations. To deploy as
many instances as possible at once, use CodeDeployDefault.AllAtOnce. To deploy half of
the instances at a time, use CodeDeployDefaultHalfAtATime. To deploy only one instance
at a time, use CodeDeployDefault.OneAtATime.",
          "enum": [
            "CodeDeployDefault.AllAtOnce",
            "CodeDeployDefault.HalfAtATime",
            "CodeDeployDefault.OneAtATime"
          ],
          "default": "CodeDeployDefault.OneAtATime"
        },
        "AutoRollbackEnabled": {
          "type": "string",
          "description": "True to enable an automatic rollback of a deployment if it
fails; if that happens, CodeDeploy redeploys the last known good revision as a new
deployment. False to not enable the automatic rollback.",
          "enum": [
            "True",
            "False"
          ],
          "default": "False"
        },
        "EC2FilterTag": {
          "type": "string",
          "description": "Key=Value pair tag for CodeDeploy to filter EC2 instances;
for example Name=Application01. The specified tag is used to identify instances as
targets for the deployment group.",
          "pattern": "^([a-zA-Z0-9\\s_.=+/-]{0,127})=([a-zA-Z0-9\\s_.=+/-]{0,255})$"
```

```
        },
        "EC2FilterTag2": {
          "type": "string",
          "description": "Second Key=Value pair tag for CodeDeploy to filter EC2
 instances; for example Environment=Test01. The specified tag is used to identify
 instances as targets for the deployment group.",
          "pattern": "^([a-zA-Z0-9\\s_.=+/-]{0,127})=([a-zA-Z0-9\\s_.=+/-]{0,255})$|^
$",
          "default": ""
        },
        "EC2FilterTag3": {
          "type": "string",
          "description": "Third Key=Value pair tag for CodeDeploy to filter EC2
 instances; for example Version=Latest. The specified tag is used to identify instances
 as targets for the deployment group.",
          "pattern": "^([a-zA-Z0-9\\s_.=+/-]{0,127})=([a-zA-Z0-9\\s_.=+/-]{0,255})$|^
$",
          "default": ""
        },
        "ServiceRoleArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of an existing CodeDeploy
 service role that grants permission to make calls to AWS services on your
 behalf, in the form arn:aws:iam::ACCOUNT_ID:role/aws-codedeploy-role. If blank
 arn:aws:iam::ACCOUNT_ID:role/aws-codedeploy-role is used.",
          "pattern": "^$|^arn:aws:iam::[0-9]{12}:role/[\\w-]+$",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "ApplicationName",
          "DeploymentConfigName",
          "AutoRollbackEnabled",
          "EC2FilterTag",
          "EC2FilterTag2",
          "EC2FilterTag3",
          "ServiceRoleArn"
        ]
      },
      "required": [
        "ApplicationName",
        "EC2FilterTag"
      ],
```

```
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-00zr0b0ozlcn3

Classifications:

- Management | Advanced stack components | S3 storage | Receive replication replica

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Receive Replication Replica",
  "description": "Receive S3 object replicas in the destination bucket.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ReceiveReplicationReplica.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ReceiveReplicationReplica"
```

```
      ],
      "default": "AWSManagedServices-ReceiveReplicationReplica"
    },
    "Region": {
      "description": "The AWS Region in which the destination account is located, in
the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DestinationBucketName": {
          "description": "The destination S3 bucket name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z0-9]$",
            "minLength": 3,
            "maxLength": 63
          },
          "maxItems": 1
        },
        "SourceBucketName": {
          "description": "The source S3 bucket name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z0-9]$",
            "minLength": 3,
            "maxLength": 63
          },
          "maxItems": 1
        },
        "ReplicationRole": {
          "description": "The ARN of the role that allows S3 to perform the replication
on your behalf.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::[0-9]{12}:role/[A-Za-z0-9_\\-/]+$"
          },
          "maxItems": 1
        },
```

```
        "EncryptReplicaKMSKey": {
          "description": "The KMS key used to encrypt destination objects.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
            "default": ""
          },
          "maxItems": 1
        },
        "OwnerTranslation": {
          "description": "True to change replica ownership to the AWS account that owns
 the destination bucket, false to not change replica ownership. This parameter cannot
 be left blank.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ],
            "default": "false"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "DestinationBucketName",
          "SourceBucketName",
          "ReplicationRole",
          "EncryptReplicaKMSKey",
          "OwnerTranslation"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DestinationBucketName",
        "SourceBucketName",
        "ReplicationRole"
      ]
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0176f0n99vcps

Classifications:

- Deployment | Advanced stack components | Tag | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Resource Tags (Review Required)",
  "description": "Add tags to existing, supported resources except those in
 AMS infrastructure stacks (stacks named mc-*). Tags simplify categorization,
 identification and targeting AWS resources. For Autoscaling, EC2, Elastic Load
 Balancing, RDS resources and S3 buckets, use the automated CT ct-3cx7we852p3af.",
  "type": "object",
  "properties": {
    "Resources": {
      "description": "Parameters for up to fifty resources that you want to tag.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "ResourceArn": {
            "description": "The ARN or the resource ID of the resource to be tagged.
 Resource ID is allowed only for these resource types: EC2 instance, EBS volume, EBS
 snapshot, AMI, and security group. All other resource types must be provided with the
 full ARN.",
```

```
            "type": "string",
            "pattern": "^arn:aws:(|[a-z][a-z0-9-]+):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-
z]+-\\d{1}):(|[0-9]{12}):([^,\\s]+)$|^(ami|i|vol|sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "AddOrUpdateTags": {
            "description": "Up to fifty tags (key/value pairs) to categorize the
 resource. If the tag exists, the value for it is overwritten. If the tag does not
 exist, it is added to the resource. Characters allowed in tags can vary by AWS
 service. For information about what characters can be used to tag resources in
 a particular AWS service, please refer to its documentation. In general, allowed
 characters in tags are letters, numbers, spaces and the following characters: _ . : /
 = + - @.",
            "type": "array",
            "items": {
              "type": "object",
              "properties": {
                "Key": {
                  "type": "string",
                  "pattern": "^(?![aA][mMwW][sS]:)[a-zA-Z0-9\\s_.:/=+\\\\\\-@\\]*]+$",
                  "minLength": 1,
                  "maxLength": 127
                },
                "Value": {
                  "type": "string",
                  "pattern": "^[a-zA-Z0-9\\s_.:/=+\\\\\\-@\\]*]+$",
                  "minLength": 1,
                  "maxLength": 255
                }
              },
              "additionalProperties": false,
              "metadata": {
                "ui:order": [
                  "Key",
                  "Value"
                ]
              },
              "required": [
                "Key",
                "Value"
              ]
            },
            "minItems": 1,
            "maxItems": 50,
            "uniqueItems": true
```

```
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "ResourceArn",
            "AddOrUpdateTags"
          ]
        },
        "required": [
          "ResourceArn",
          "AddOrUpdateTags"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Resources",
      "Priority"
    ]
  },
  "required": [
    "Resources"
  ]
}
```

Schema for Change Type ct-01zl37gmuk4q2

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Delete SAML identity provider


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete SAML Identity Provider",
  "description": "Delete a SAML identity provider (IdP). The given IdP must not be
 referenced in any IAM role and must not be the only IdP in the account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleDeleteSamlProvider-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleDeleteSamlProvider-Admin"
      ],
      "default": "AWSManagedServices-HandleDeleteSamlProvider-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Name": {
          "description": "The name of the SAML IdP.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[\\w._-]{1,128}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "MetadataBackup": {
```

```
          "description": "True for a backup of the SAML provider metadata to be taken
 before deleting, False for no backup to be taken. Default is True.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "MetadataBackup"
        ]
      },
      "required": [
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-02ocqy2i0jx3t

Classifications:

- Management | Advanced stack components | RDS database stack | Start Aurora cluster


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start Aurora DB Cluster",
  "description": "Start an Aurora DB cluster, which is a provisioned capacity type
 and does not have cross-region read replicas. The cluster must be in the 'stopped'
 state.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartDBCluster.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartDBCluster"
      ],
      "default": "AWSManagedServices-StartDBCluster"
    },
    "Region": {
      "description": "The AWS Region where the cluster is.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
```

```
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBClusterIdentifier": {
          "description": "The unique RDS DB cluster identifier.",
          "type": "string",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "DBClusterIdentifier"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBClusterIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-02u0hoaa9grat

Classifications:

- Management | Standard stacks | Stack | Reboot

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Reboot stack",
  "description": "Use to reboot all running EC2 and RDS DB instances in the specified
 stack.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "StackId": {
      "pattern": "^stack-[a-z0-9]{17}$",
      "description": "The ID of the stack to reboot, in the form stack-
a1b2c3d4e5f67890e. All running EC2 and RDS DB instances in the stack are rebooted.",
      "type": "string"
    }
  },
  "required": [
    "StackId"
  ]
}
```

Schema for Change Type ct-03ms1d7xrck8w

Classifications:

- Management | Advanced stack components | EC2 instance stack | Update termination protection

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Termination Protection",
  "description": "Update existing defined termination protection for EC2 instances.",
```

```
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
ManageResourceTerminationProtection.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ManageResourceTerminationProtection"
      ],
      "default": "AWSManagedServices-ManageResourceTerminationProtection"
    },
    "Region": {
      "description": "The AWS Region in which the EC2 instance is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ResourceId": {
          "description": "EC2 instance ID.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8,17}$"
          },
          "maxItems": 1
        },
        "TerminationProtectionDesiredState": {
          "description": "Enabled to protect your instance against elimination.
 Disabled to allow your instance to be eliminated.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "enabled",
              "disabled"
            ]
          },
          "maxItems": 1
        }
      },
      "metadata": {
```

```
        "ui:order": [
          "ResourceId",
          "TerminationProtectionDesiredState"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResourceId",
        "TerminationProtectionDesiredState"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-03t7kvuwx6rgr

Classifications:

- Management | Advanced stack components | EC2 instance stack | Start

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start EC2 Instances",
  "description": "Start up to 50 stopped EC2 instances.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartInstances.",
      "type": "string",
```

```
      "enum": [
        "AWSManagedServices-StartInstances"
      ],
      "default": "AWSManagedServices-StartInstances"
    },
    "Region": {
      "description": "The AWS Region where the instances are, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceIds": {
          "description": "A list of up to 50 EC2 instance IDs, in the form
i-1234567890abcdef0 or i-b188560f.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
```

```
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-03ytgoevfebjr

Classifications:

- Management | Directory Service | DNS | Update cluster permissions

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Cluster Permissions",
  "description": "Grants full control to the Cluster object on the Listener object to
 bring the SQL Server Listener object online. For multi-account landing zone (MALZ),
 use this change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateClusterDNSPermission-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateClusterDNSPermission-Admin"
      ],
      "default": "AWSManagedServices-UpdateClusterDNSPermission-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterName": {
          "description": "The name of the Cluster record in DNS.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_\\-]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ClusterNodeComputerName": {
          "description": "The name of the Cluster object that is granted permissions to
 the Cluster DNS record.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_\\-]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ClusterName",
          "ClusterNodeComputerName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ClusterName",
        "ClusterNodeComputerName"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-042luqo63j4mx

Classifications:

- Management | AMS Resource Scheduler | Period | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Resource Scheduler Period",
  "description": "Delete an existing period used in AMS Resource Scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteScheduleOrPeriod.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteScheduleOrPeriod"
      ],
      "default": "AWSManagedServices-DeleteScheduleOrPeriod"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ConfigurationType": {
          "description": "Specify the value: period. This explicitly requests that the
Resource Scheduler period be deleted. The option cannot be left blank; it must be
period.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "period"
            ],
```

```
            "default": "period"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Name": {
          "description": "The name of the period to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,64}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ConfigurationType",
          "Name"
        ]
      },
      "required": [
        "ConfigurationType",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-046aizcwg5idf

Classifications:

- Deployment | Advanced stack components | AMI | Copy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Copy AMI",
  "description": "Copy an Amazon Machine Image (AMI) in your AMS account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CopyAMI.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CopyAMI"
      ],
      "default": "AWSManagedServices-CopyAMI"
    },
    "Region": {
      "description": "The AWS Region to copy the AMI to, in the form us-east-1. This
 must be the account's default AWS Region.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Name": {
          "description": "A name for the new AMI.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[A-Za-z0-9\\-\\/\\(\\)_.\\ ]{3,128}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SourceImageId": {
          "description": "The ID of the AMI to copy.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^ami-[a-f0-9]{8}$|^ami-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SourceRegion": {
          "description": "The ID of the AWS Region that contains the source AMI, in the
form us-east-1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Encrypted": {
          "description": "True to encrypt the snapshot of the destination AMI. The
default customer master key (CMK) for Amazon Elastic Block Store (EBS) is used unless
you specify a non-default AWS Key Management Service (KMS) CMK using the KmsKeyId
parameter. False to not encrypt the snapshot. Default is False.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KmsKeyId": {
          "description": "The KMS key to encrypt the snapshot of the destination AMI.
Specify the KMS Key ARN or the KMS key identifier. If left blank and the snapshot of
the source AMI is encrypted, the snapshot of the target AMI is encrypted using the
default EBS KMS key.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
```

```
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "SourceImageId",
          "SourceRegion",
          "Encrypted",
          "KmsKeyId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "Name",
        "SourceImageId",
        "SourceRegion"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-04gzyy008v1bg

Classifications:

- Management | Advanced stack components | KMS alias | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete KMS Alias",
  "description": "Delete an alias of an AWS Key Management Service (KMS) customer
 master key (CMK).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteKMSAlias.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteKMSAlias"
      ],
      "default": "AWSManagedServices-DeleteKMSAlias"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AliasName": {
          "description": "Name of the alias to be deleted. Do not specify the prefix
 alias/, it will be added during the execution.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!alias/)(?!aws/)[a-zA-Z0-9/_-]{1,250}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
```

```
      "metadata": {
        "ui:order": [
          "AliasName"
        ]
      },
      "required": [
        "AliasName"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-059ewa92tc2i1

Classifications:

- Management | Advanced stack components | EBS snapshot | Archive

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Archive EBS Snapshots",
  "description": "Archive Elastic Block Store (EBS) snapshots. The maximum number of
 EBS snapshots that can be archived concurrently depends on the 'In-progress snapshot
 archives per account' AWS Service Quota. Snapshots that are in the 'completed'
 state, storage tier is 'standard', or belonging to the current owner account, can be
 archived. Snapshots created by the AWS Backup service, used by AMIs, or shared with
 other accounts, cannot be archived. If you specify snapshots that are invalid, or the
 archival in-progress quota limit is reached, the RFC fails.",
```

```
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ArchiveEBSSnapshots.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ArchiveEBSSnapshots"
      ],
      "default": "AWSManagedServices-ArchiveEBSSnapshots"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SnapshotIds": {
          "description": "A comma-separated list of the EBS snapshots to archive. The
maximum number of in-progress snapshot archives per account can be checked through the
AWS Service Quotas console (search: In-progress snapshot archives per account).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 100
        }
      },
      "metadata": {
        "ui:order": [
          "SnapshotIds"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SnapshotIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
```

```
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-05muqzievnxk5

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create target endpoint (S3)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS target endpoint for S3",
  "description": "Use to create a Database Migration Service (DMS) target endpoint for
 S3.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
```

```
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-knghtmmgefafdq89u",
      "type": "string",
```

```
      "enum": [
        "stm-knghtmmgefafdq89u"
      ],
      "default": "stm-knghtmmgefafdq89u"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "EndpointIdentifier": {
          "type": "string",
          "description": "The identifier to be used for the target endpoint. This is a
label for the endpoint to help you identify it. It must be unique for all endpoints
owned by your AWS account in the current region. It must begin with a letter, must
contain only ASCII letters, digits and hyphens and must not end with a hyphen or
contain two consecutive hyphens.",
          "pattern": "^$|(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$",
          "default": ""
        },
        "EngineName": {
          "type": "string",
          "description": "Must be S3.",
          "enum": [
            "s3"
          ],
          "default": "s3"
        },
        "ExtraConnectionAttributes": {
          "type": "string",
          "description": "Additional attributes associated with the connection. For
example, to specify a maximum file size of 512 KB of any CSV file created while
migrating to S3 specify maxFileSize=512. See 'Targets for Data Migration' in AWS DMS
documentation.",
          "default": ""
        },
        "S3BucketFolder": {
```

```
          "type": "string",
          "description": "The folder name in the S3 bucket. If provided, tables
are created in the path <bucketFolder>/<schema_name>/<table_name>/ instead of
<schema_name>/<table_name>/ within the bucket.",
          "default": ""
        },
        "S3BucketName": {
          "type": "string",
          "description": "The name of the S3 bucket for the target endpoint. Must be in
the same region as the DMS replication instance you are using to migrate data."
        },
        "S3CompressionType": {
          "type": "string",
          "description": "If, and how, target files should be compressed. Use GZIP to
compress the target files in the target endpoint. Use NONE for no file compression.",
          "enum": [
            "GZIP",
            "NONE"
          ],
          "default": "NONE"
        },
        "S3CsvDelimiter": {
          "type": "string",
          "description": "The delimiter used to separate columns in the target files.
Leave blank to use the default comma (,) delimiter.",
          "default": ""
        },
        "S3CsvRowDelimiter": {
          "type": "string",
          "description": "The delimiter used to separate rows in the source files.
Leave blank to use the default carriage return (\\n) delimiter.",
          "default": ""
        },
        "S3ServiceAccessRoleArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the service access IAM
role.",
          "pattern": "^$|^arn:aws:iam::[0-9]{12}:role/[\\w-]+$"
        }
      },
      "metadata": {
        "ui:order": [
          "EndpointIdentifier",
          "EngineName",
```

```
          "ExtraConnectionAttributes",
          "S3BucketFolder",
          "S3BucketName",
          "S3CompressionType",
          "S3CsvDelimiter",
          "S3CsvRowDelimiter",
          "S3ServiceAccessRoleArn"
        ]
      },
      "required": [
        "EngineName",
        "S3BucketName",
        "S3ServiceAccessRoleArn"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-05yb337abq3x5

Classifications:

- Management | Advanced stack components | KMS key | Share (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Share KMS Key",
  "description": "Allow cross-account access to a KMS key by adding a statement to the
 key policy with encrypt and decrypt permissions.",
  "type": "object",
  "properties": {
    "KMSKeyArn": {
      "description": "The Amazon Resource Name (ARN) of the KMS key, in the form
 arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab.",
      "type": "string",
      "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-f0-9]{4}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$"
    },
    "TargetAccountId": {
      "description": "The ID of the AWS account that you want to share the KMS key
 with.",
      "type": "string",
      "pattern": "^[0-9]{12}$"
    },
    "IncludeKeyGrantPermissions": {
      "description": "Add permissions for managing grants of the KMS key. These are
 required for performing tasks such as copying an encrypted AMI or snapshot.",
      "type": "boolean",
      "default": false
    },
    "IAMUserOrRoleARN": {
      "description": "The ARN of an IAM Role or User in the target account to grant
 permission to. If no value is provided, the root principal of the target account is
 used.",
      "type": "string",
      "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:(role|user)/[A-Za-
z0-9_-]+$|^$",
      "default": ""
    },
    "Priority": {
```

```
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "KMSKeyArn",
      "IncludeKeyGrantPermissions",
      "TargetAccountId",
      "IAMUserOrRoleARN",
      "Priority"
    ]
  },
  "required": [
    "KMSKeyArn",
    "TargetAccountId"
  ]
}
```

Schema for Change Type ct-063qsm82cfxu6

Classifications:

- Deployment | Advanced stack components | EBS Volume | Create from backup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EBS From Backup",
  "description": "Create an AWS Elastic Block Store (EBS) stack from backup.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartRestoreJobEBS.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-StartRestoreJobEBS"
      ],
      "default": "AWSManagedServices-StartRestoreJobEBS"
    },
    "Region": {
      "description": "The AWS Region in which the EBS snapshot is located, in the form
us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AvailabilityZone": {
          "description": "The Availability Zone in which to restore the EBS snapshot,
in the form us-east-1a.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-z]{2}((-gov))?-[a-z]+-[0-9]{1}[a-z]{1})$"
          },
          "maxItems": 1
        },
        "BackupVaultName": {
          "description": "The name of a logical container where backups are stored.
The backup vault name is case sensitive and must contain from 2 to 50 alphanumeric
characters or hyphens.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "maxItems": 1
        },
        "IOPS": {
          "description": "The requested number of I/O operations per second that the
new EBS volume can support if VolumeType is io1, io2 or gp3. This value is ignored
for other volume types. If VolumeType is gp3, then the IOPS should be between 3000 and
16000, else it should be between 100 and 64000. The IOPS must respect the max ratio of
50 IOPS per GiB.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3]
[0-9]{3}|64000)$"
          },
          "maxItems": 1
        },
        "Throughput": {
          "description": "The Throughput to use for the restored volume if VolumeType
 is gp3. If VolumeType is not gp3, any value provided here is ignored. The Throughput
 should be between 125 and 1000.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
          },
          "maxItems": 1
        },
        "RecoveryPointArn": {
          "description": "The Amazon Resource Name (ARN) that uniquely identifies the
 recovery point to restore.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        },
        "VolumeSize": {
          "description": "The size of the volume, in GiBs. The volume size must be
 equal to or larger than the snapshot size. If not specified, the default will be the
 snapshot size. Valid values are between 1 and 16384.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|1[0-5][0-9]{3}|16[0-2][0-9]{2}|
163[0-7][0-9]|1638[0-4])$"
          },
          "maxItems": 1
        },
        "VolumeType": {
          "description": "The volume type for the restored volume. Choose io1, io2, gp2
 or gp3 for SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1
 for HDD-backed volumes optimized for large streaming workloads. Choose standard for
```

```
HDD-backed volumes suitable for workloads where data is infrequently accessed. If not
specified gp3 will be used as default.",
        "type": "array",
        "items": {
          "type": "string",
          "default": "gp3",
          "pattern": "^(standard|io1|io2|gp2|gp3|sc1|st1)$"
        },
        "maxItems": 1
      }
    },
    "metadata": {
      "ui:order": [
        "AvailabilityZone",
        "BackupVaultName",
        "IOPS",
        "Throughput",
        "RecoveryPointArn",
        "VolumeSize",
        "VolumeType"
      ]
    },
    "additionalProperties": false,
    "required": [
      "AvailabilityZone",
      "BackupVaultName",
      "RecoveryPointArn"
    ]
  }
},
"metadata": {
  "ui:order": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
},
"additionalProperties": false,
"required": [
  "DocumentName",
  "Region",
  "Parameters"
]
```

```
}
```

Schema for Change Type ct-06bwg93ukgg8t

Classifications:

- Deployment | Advanced stack components | VPC | Add static route (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Static Route",
  "description": "Create a static route on your route table inside a VPC.",
  "type": "object",
  "properties": {
    "RouteTableId": {
      "description": "The ID of the route table for the route, in the form of
 rtb-01234567890abcdef.",
      "type": "string",
      "pattern": "^rtb-[a-z0-9]{8,17}$"
    },
    "Destination": {
      "description": "The IPv4 CIDR address block in the form 192.168.10.0/24 or the ID
 of a prefix list in the form pl-01234567890abcdef used for the destination match.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/(3[0-2]|[1-2][0-9]|[0-9]))$|^pl-[a-z0-9]{8,17}$"
    },
    "RouteTableTarget": {
      "description": "The ID of the resource that will serve as the route table target.
 You must specify one of the following targets: internet gateway or virtual private
 gateway, NAT gateway or VPC peering connection.",
      "type": "string",
      "pattern": "^(vgw|igw|nat|tgw|pcx)-[a-z0-9]{8,17}$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "RouteTableId",
      "Destination",
      "RouteTableTarget",
      "Priority"
    ]
  },
  "required": [
    "RouteTableId",
    "Destination",
    "RouteTableTarget"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-06mjngx5flwto

Classifications:

- Deployment | Standard stacks | High availability two-tier stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create high availability two-tier stack",
  "description": "Creates a stack consisting of an Auto Scaling group, an RDS DB
 instance, and a load balancer (ELB). Optionally allows for application deployment with
 CodeDeploy by also creating a CodeDeploy application and deployment group both named
 the value given for ApplicationName. All resource parameters can be configured.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "TimeoutInMinutes": {
```

```
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "default": 360
    },
    "VpcId": {
      "description": "The ID of the VPC to create the Auto Scaling group in, in the
form vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack; this becomes the searchable stack name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to forty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
```

```
      "maxItems": 40,
      "uniqueItems": true
    },
    "AutoScalingGroup": {
      "description": "Specifications for the application tier.",
      "type": "object",
      "properties": {
        "AmiId": {
          "description": "The AMI ID for the Auto Scaling Group to utilize, in the form
ami-0123abcd or ami-01234567890abcdef.",
          "type": "string",
          "pattern": "^ami-[a-z0-9]{8}$|^ami-[a-z0-9]{17}$"
        },
        "Cooldown": {
          "description": "The number of seconds after a scaling activity is completed
before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600,
          "default": 300
        },
        "DesiredCapacity": {
          "description": "The number of EC2 instances you want running in the group.
This number must be greater than or equal to the MinInstances setting and less than or
equal to the MaxInstances setting.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 2
        },
        "EBSOptimized": {
          "description": "True to create EBS-optimized instances, false to not.
EBS-optimization provides dedicated throughput to Amazon EBS and optimal EBS I/O
performance.",
          "type": "boolean",
          "default": false
        },
        "HealthCheckGracePeriod": {
          "description": "The amount of time, in seconds, that Auto Scaling waits
before checking the health status of an EC2 instance that has come into service.
During this time, any health check failures for the instance are ignored.",
          "type": "integer",
          "minimum": 600,
          "maximum": 1800,
```

```
          "default": 1800
        },
        "IAMInstanceProfile": {
          "description": "The IAM instance profile for the Auto Scaling group. EC2
instances launched with an IAM role automatically have AWS security credentials
available.",
          "type": "string",
          "default": "customer-mc-ec2-instance-profile"
        },
        "InstanceDetailedMonitoring": {
          "description": "True to enable detailed monitoring on the instances in the
Auto Scaling group, false to use only basic monitoring.",
          "type": "boolean",
          "default": true
        },
        "InstanceRootVolumeIops": {
          "description": "The Iops to use for the root volume if io1 volume type is
specified.",
          "type": "integer",
          "minimum": 0,
          "maximum": 20000,
          "default": 0
        },
        "InstanceRootVolumeName": {
          "description": "The name of the root volume to use. Defaults to /dev/xvda for
Linux, and /dev/sda for Windows.",
          "type": "string"
        },
        "InstanceRootVolumeSize": {
          "description": "The size of the root volume for the instance. Defaults to 20
GiB for Linux, and 60 GiB for Windows.",
          "type": "integer",
          "minimum": 8,
          "maximum": 16000
        },
        "InstanceRootVolumeType": {
          "description": "Choose io1 or gp2 for SSD-backed volumes optimized for
transactional workloads; choose standard for HDD-backed volumes optimized for large
streaming workloads.",
          "type": "string",
          "enum": [
            "standard",
            "io1",
            "gp2"
```

```
          ],
          "default": "standard"
        },
        "InstanceType": {
          "description": "The instance type for the Auto Scaling group to use when
creating new EC2 instances.",
          "type": "string",
          "default": "m4.large"
        },
        "MaxInstances": {
          "description": "The maximum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 2
        },
        "MinInstances": {
          "description": "The minimum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 2
        },
        "ScaleDownPolicyCooldown": {
          "description": "The number of seconds after a scale-down activity is
completed before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600,
          "default": 300
        },
        "ScaleDownPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ScaleMetricName threshold.",
          "type": "integer",
          "minimum": 2,
          "default": 4
        },
        "ScaleDownPolicyPeriod": {
          "description": "The time over which the specified ScaleDownPolicyStatistic is
applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
```

```
          "multipleOf": 60,
          "minimum": 60,
          "default": 60
        },
        "ScaleDownPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale down.",
          "type": "integer",
          "maximum": 0,
          "default": -1
        },
        "ScaleDownPolicyStatistic": {
          "description": "The statistic to apply to the alarm's ScaleMetricName.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ],
          "default": "Average"
        },
        "ScaleDownPolicyThreshold": {
          "description": "The value against which the specified
ASGScaleDownPolicyStatistic is compared.",
          "type": "number",
          "default": 35
        },
        "ScaleMetricName": {
          "description": "The metric to use in a scaling event. Exceeding the metric
triggers an alarm.",
          "type": "string",
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
```

```
            "StatusCheckFailed_System"
          ],
          "default": "CPUUtilization"
        },
        "ScaleUpPolicyCooldown": {
          "description": "The amount of time, in seconds, after a scale-up activity is
completed before any further trigger-related scaling activities can start.",
          "type": "integer",
          "minimum": 60,
          "default": 60
        },
        "ScaleUpPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ScaleMetricName threshold.",
          "type": "integer",
          "minimum": 2,
          "default": 2
        },
        "ScaleUpPolicyPeriod": {
          "description": "The time over which the specified ScaleUpPolicyStatistic is
applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60,
          "default": 60
        },
        "ScaleUpPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale up.",
          "type": "integer",
          "minimum": 0,
          "default": 2
        },
        "ScaleUpPolicyStatistic": {
          "description": "The statistic to apply to the alarm's ScaleMetricName.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ],
          "default": "Average"
        },
```

```
        "ScaleUpPolicyThreshold": {
          "description": "The value against which the specified ScaleUpPolicyStatistic
is compared.",
          "type": "number",
          "default": 75
        },
        "SubnetIds": {
          "description": "One or more subnets for the Auto Scaling group to launch
instances into (scale up) or remove instances from (scale down), in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 2,
          "uniqueItems": true
        },
        "UserData": {
          "description": "A comma-delimited list where each element is a line of script
to be run on boot.",
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 1,
          "default": [
            ""
          ]
        }
      },
      "additionalProperties": false,
      "required": [
        "AmiId",
        "SubnetIds"
      ]
    },
    "LoadBalancer": {
      "description": "Specifications for the load-balancing tier.",
      "type": "object",
      "properties": {
        "SubnetIds": {
```

```
          "description": "One or more subnet IDs for the load balancer, in the form
 subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "HealthCheckInterval": {
          "description": "The approximate interval, in seconds, between health
 checks.",
          "type": "number",
          "minimum": 5,
          "maximum": 300,
          "default": 30
        },
        "HealthCheckTarget": {
          "description": "Specifies the instance being checked. The protocol can be
 TCP, HTTP, HTTPS, or SSL.  The range of valid ports is 1 through 65535. For example,
 HTTP:80/",
          "type": "string",
          "pattern": "^(HTTP|HTTPS):[0-9]{1,5}[/][a-zA-Z0-9/_.-]*$|^(SSL|TCP):[0-9]
{1,5}$"
        },
        "HealthCheckTimeout": {
          "description": "The amount of time, in seconds, to wait for a response to a
 health check. Must be less than the value for HealthCheckInterval.",
          "type": "number",
          "minimum": 2,
          "maximum": 60,
          "default": 5
        },
        "Public": {
          "description": "True if the load balancer endpoint is public, false if it
 is not. Default is false. Set to true if you choose a public subnet for the load
 balancer.",
          "type": "boolean",
          "default": false
        },
        "AccessCIDRRange": {
          "default": "0.0.0.0/0",
```

```
          "description": "IPv4 CIDR block that the load balancer can receive traffic
from.",
          "type": "string"
        }
      },
      "additionalProperties": false,
      "required": [
        "SubnetIds"
      ]
    },
    "Database": {
      "description": "Specifications for the RDS DB instance.",
      "type": "object",
      "properties": {
        "AllocatedStorage": {
          "description": "The amount of storage (in gigabytes) to be initially
allocated for the database (DB) instance.",
          "type": "number",
          "minimum": 5,
          "maximum": 6144
        },
        "BackupRetentionPeriod": {
          "description": "The number of days for which automatic DB snapshots are
retained.  Setting this to a positive number enables backups. Setting this to 0
disables automated backups.",
          "type": "number",
          "minimum": 0,
          "maximum": 35,
          "default": 7
        },
        "Backups": {
          "description": "True if the RDS instance should have automatic backups, false
if it should not. Default is true.",
          "type": "boolean",
          "default": true
        },
        "DBEngine": {
          "description": "The name of the database engine for the DB instance. Not
every database engine is available for every AWS region.",
          "type": "string",
          "enum": [
            "MySQL",
            "oracle-se1",
            "oracle-se",
```

```
            "oracle-ee",
            "sqlserver-ee",
            "sqlserver-se",
            "sqlserver-ex",
            "sqlserver-web",
            "postgres"
          ]
        },
        "DBName": {
          "default": "main",
          "description": "A name for the database. The meaning of this parameter
 differs according to the database engine you use.",
          "type": "string",
          "minLength": 1
        },
        "EngineVersion": {
          "description": "The version number of the database engine to use.",
          "type": "string"
        },
        "InstanceType": {
          "description": "The compute and memory capacity for the DB instance.",
          "type": "string",
          "enum": [
            "db.m1.medium",
            "db.m1.large",
            "db.m1.xlarge",
            "db.m2.xlarge",
            "db.m2.2xlarge",
            "db.m2.4xlarge",
            "db.m3.medium",
            "db.m3.large",
            "db.m3.xlarge",
            "db.m3.2xlarge",
            "db.r3.large",
            "db.r3.xlarge",
            "db.r3.2xlarge",
            "db.r3.4xlarge",
            "db.r3.8xlarge",
            "db.t2.micro",
            "db.t2.small",
            "db.t2.medium"
          ],
          "default": "db.m3.medium"
        },
```

```
      "IOPS": {
        "description": "The provisioned IOPS for RDS storage. Must be a multiple
between 3 and 10 of the storage amount for the DB instance. Must also be an integer
multiple of 1000. For example, if the size of your DB instance is 500 GB, then your
Iops value can be 2000, 3000, 4000, or 5000.",
        "type": "number",
        "default": 0
      },
      "LicenseModel": {
        "description": "License model information for this DB instance.",
        "type": "string",
        "enum": [
          "bring-your-own-license",
          "general-public-license",
          "license-included",
          "postgresql-license"
        ]
      },
      "MasterUsername": {
        "description": "The username that you will use with the configured
MasterUserPassword to log in to your DB instance. Must begin with a letter and contain
only alphanumeric characters.",
        "type": "string",
        "pattern": "^[a-zA-Z][a-zA-Z0-9]{1,127}$"
      },
      "MasterUserPassword": {
        "description": "The password that you will use with the configured
MasterUserName to log in to your DB instance. Must contain from 8 to 30 printable
ASCII alphanumeric characters (excluding backslash, double quotes, and at sign).",
        "type": "string",
        "pattern": "^[!#-.0-?A-~]{8,30}$",
        "metadata": {
          "ams:sensitive": true
        }
      },
      "MultiAZ": {
        "description": "True to have a standby replica of your DB instance created in
another Availability Zone for failover support, false to not have a standby replica.
Default is true.",
        "type": "boolean",
        "default": true
      },
      "PreferredBackupWindow": {
```

```
          "description": "The daily time range during which automated backups are
 created if BackupRetentionPeriod is set to a positive number. Must be in the format
 hh:mm-hh:mm (24-hour format), in Universal Coordinated Time (UTC). Must not conflict
 with the PreferredMaintenanceWindow setting, and must be at least 30 minutes.",
          "type": "string",
          "default": "22:00-23:00",
          "pattern": "^(0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]-(0[0-9]|1[0-9]|2[0-3]):[0-5]
[0-9]$"
        },
        "Port": {
          "description": "The port number on which the database accepts connections.
 Defaults vary by DB engine.",
          "type": "number"
        },
        "PreferredMaintenanceWindow": {
          "description": "The weekly time range (in UTC) during which system
 maintenance can occur.",
          "type": "string",
          "default": "wed:03:32-wed:04:02",
          "pattern": "^(mon|tues|wed|thurs|fri|sat|sun):(0[0-9]|1[0-9]|2[0-3]):[0-5]
[0-9]-(mon|tues|wed|thurs|fri|sat|sun):(0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]$"
        },
        "StorageEncrypted": {
          "description": "True to enable database encryption, false to not. Default is
 false.",
          "type": "boolean",
          "default": false
        },
        "StorageEncryptionKey": {
          "description": "The ARN of the custom KMS key to encrypt the database
 if StorageEncrypted = true. If StorageEncrypted = true and you do not specify a
 StorageEncryptionKey, RDS uses your default encryption key, which AWS KMS creates.
 Your AWS account has a different default encryption key for each AWS region.",
          "type": "string",
          "default": ""
        },
        "StorageType": {
          "description": "Storage type for the RDS instance. If you specify io1, you
 must also include a value for the IOPS parameter.",
          "type": "string",
          "enum": [
            "standard",
            "gp2",
            "io1"
```

```
          ],
          "default": "gp2"
        },
        "SubnetIds": {
          "description": "Subnet IDs for the RDS instance, in the form subnet-0123abcd
or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 2,
          "maxItems": 20,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "required": [
        "DBName",
        "DBEngine",
        "EngineVersion",
        "LicenseModel",
        "MasterUsername",
        "MasterUserPassword",
        "SubnetIds"
      ]
    },
    "Application": {
      "description": "Optional parameters for including an application to deploy with
CodeDeploy. Given a unique ID if none is provided.",
      "type": "object",
      "properties": {
        "ApplicationName": {
          "description": "The name of an AWS CodeDeploy application.",
          "type": "string",
          "minLength": 1,
          "maxLength": 100,
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        },
        "DeploymentConfigName": {
          "description": "The configuration for deployment operations: as many
instances as possible at once, half of the instances at a time, or only one instance
at a time.",
          "type": "string",
```

```
          "enum": [
            "CodeDeployDefault.AllAtOnce",
            "CodeDeployDefault.HalfAtATime",
            "CodeDeployDefault.OneAtATime"
          ],
          "default": "CodeDeployDefault.OneAtATime"
        }
      },
      "additionalProperties": false
    },
    "EnforceIMDSv2": {
      "description": "For the instance to be launched with only Instance Metadata
 Service Version 2 (IMDSv2), use required; if IMDSv2 is not required, use optional.
 Default is optional.",
      "type": "string",
      "default": "optional"
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "Name",
    "LoadBalancer",
    "AutoScalingGroup",
    "Database",
    "VpcId",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-07jzw8bzd2on7

Classifications:

- Management | Monitoring and notification | GuardDuty IP set | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update GuardDuty IPSet",
  "description": "Use to update an Amazon GuardDuty IPSet instance which is a list of
 trusted IP addresses that have been whitelisted for highly secure communication with
 your AWS environment.",
```

```
 "type": "object",
 "properties": {
   "Activate": {
     "description": "Specified whether the IPSet is active or not.",
     "type": "boolean",
     "default": true
   },
   "DetectorId": {
     "description": "The detector ID that specifies the GuardDuty service to which you
want to update an IPSet. Leave this blank to use the only detector in the selected
region (this will not succeed if there is more than one detector in the selected
region).",
     "pattern": "^[a-fA-F0-9]{32}$|^$",
     "type": "string"
   },
   "IpSet": {
     "description": "The URI of the file that contains the IPSet.",
     "minLength": 1,
     "type": "string"
   },
   "IpSetId": {
     "description": "The unique ID that specifies the IPSet that you want to
update.",
     "type": "string",
     "minLength": 1
   },
   "Name": {
     "description": "The friendly name to identify the IPSet. This name is displayed
in all findings that are triggered by activity that involves IP addresses included in
this IPSet.",
     "minLength": 1,
     "type": "string"
   },
   "Region": {
     "description": "The region containing the GuardDuty detector to use; in the form
of us-east-1.",
     "minLength": 1,
     "type": "string"
   },
   "Priority": {
     "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
     "type": "string",
     "enum": [
```

```
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "IpSetId",
      "Name",
      "IpSet",
      "Activate",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "IpSetId",
    "Region"
  ]
}
```

Schema for Change Type ct-08avsj2e9mc7g

Classifications:

- Deployment | Monitoring and notification | GuardDuty IP set | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create GuardDuty IPSet",
  "description": "Use to create an Amazon GuardDuty IPSet instance which is a list of
 trusted IP addresses that have been whitelisted for highly secure communication with
 your AWS environment.",
  "type": "object",
  "properties": {
    "Activate": {
      "description": "Specified whether the IPSet is active or not.",
      "type": "boolean",
      "default": true
```

```
    },
    "DetectorId": {
      "description": "The detector ID that specifies the GuardDuty service to which you
want to add an IPSet. Leave this blank to use the only detector in the selected region
(this will not succeed if there is more than one detector in the selected region).",
      "pattern": "^[a-fA-F0-9]{32}$|^$",
      "type": "string"
    },
    "Format": {
      "default": "TXT",
      "description": "The format of the file that contains the IPSet.",
      "enum": [
        "TXT",
        "STIX",
        "OTX_CSV",
        "ALIEN_VAULT",
        "PROOF_POINT",
        "FIRE_EYE"
      ],
      "type": "string"
    },
    "Name": {
      "description": "The friendly name to identify the IPSet. This name is displayed
in all findings that are triggered by activity that involves IP addresses included in
this IPSet.",
      "minLength": 1,
      "type": "string"
    },
    "IpSet": {
      "description": "The URI of the file that contains the IPSet.",
      "minLength": 1,
      "type": "string"
    },
    "Region": {
      "description": "The region containing the GuardDuty detector to use; in the form
of us-east-1.",
      "minLength": 1,
      "type": "string"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
```

```
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Name",
      "IpSet",
      "Format",
      "Activate",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Name",
    "IpSet",
    "Region"
  ]
}
```

Schema for Change Type ct-09qbhy7kvtxqw

Classifications:

- Management | Advanced stack components | EC2 instance stack | Reboot

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Reboot EC2 instance",
  "description": "Use to reboot an EC2 instance.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "InstanceId": {
      "pattern": "^i-[a-zA-Z0-9]{8}$|^i-[a-zA-Z0-9]{17}$",
      "description": "ID of the instance to reboot, in the form i-12345678901234567 or
 i-1234567.",
```

```
      "type": "string"
    }
  },
  "required": [
    "InstanceId"
  ]
}
```

Schema for Change Type ct-09t6q7j9v5hrn

Classifications:

- Deployment | Standard stacks | High availability one-tier stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create high availability one-tier stack",
  "description": "Use to create an Application Load Balancer and an Auto Scaling
 Group.",
  "type": "object",
  "properties": {
    "DatabaseStackId": {
      "description": "Stack ID of the database to use, in the form
 stack-1ab2cd3456789101.",
      "type": "string",
      "pattern": "^stack-[0-9a-z]{17}$"
    },
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to forty tags (key/value pairs) to categorize the resource.",
```

```
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "default": 360
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "ApplicationLoadBalancer": {
      "description": "Specifications for the ALB.",
      "type": "object",
      "properties": {
```

```
      "HealthCheckHealthyThreshold": {
        "description": "The number of consecutive health check successes required to
declare an EC2 instance healthy.",
        "type": "number",
        "minimum": 2,
        "maximum": 10,
        "default": 2
      },
      "HealthCheckIntervalInSeconds": {
        "description": "The amount of time, in seconds, between health checks.",
        "type": "number",
        "minimum": 5,
        "maximum": 300,
        "default": 10
      },
      "HealthCheckTargetPath": {
        "default": "/",
        "description": "The ping path destination on the application hosts where the
load balancer sends health check requests.",
        "type": "string"
      },
      "HealthCheckTargetPort": {
        "description": "The port the load balancer uses when performing health checks
on targets. The default is traffic-port, which indicates the port on which each target
receives traffic from the load balancer.",
        "type": "number",
        "minimum": 1,
        "maximum": 65535
      },
      "HealthCheckTargetProtocol": {
        "default": "HTTP",
        "description": "The protocol the load balancer uses when performing health
checks on targets.",
        "type": "string",
        "enum": [
          "HTTP",
          "HTTPS"
        ]
      },
      "HealthCheckTimeoutSeconds": {
        "description": "The amount of time, in seconds, to wait for a response to a
health check. Must be less than the value for HealthCheckIntervalInSeconds.",
        "type": "number",
        "minimum": 2,
```

```
          "maximum": 60,
          "default": 5
        },
        "HealthCheckUnhealthyThreshold": {
          "description": "The number of consecutive health check failures required to
 declare an EC2 instance unhealthy.",
          "type": "number",
          "minimum": 2,
          "maximum": 10,
          "default": 2
        },
        "InstancePort": {
          "default": 80,
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "type": "number",
          "minimum": 1,
          "maximum": 65535
        },
        "InstanceProtocol": {
          "default": "HTTP",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "TCP"
          ]
        },
        "LoadBalancerCookieExpirationPeriodInSeconds": {
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "type": "number"
        },
        "LoadBalancerPort": {
          "default": 80,
          "description": "The port number for the load balancer to use when routing
 external incoming traffic.",
          "type": "number",
          "minimum": 1,
          "maximum": 65535
        },
```

```
        "LoadBalancerAccessCIDRRange": {
          "default": "0.0.0.0/0",
          "description": "IPv4 CIDR block that the load balancer can receive traffic
from.",
          "type": "string"
        },
        "LoadBalancerProtocol": {
          "default": "HTTP",
          "description": "The transport protocol to use for routing front-end
connections (client to load balancer).",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS"
          ]
        },
        "LoadBalancerSslPolicy": {
          "default": "ELBSecurityPolicy-2016-08",
          "description": "The security policy that defines the ciphers and protocols
that the load balancer supports. Only applies if ALBLoadBalancerProtocol = HTTPS.",
          "type": "string",
          "enum": [
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2015-05",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10"
          ]
        },
        "Public": {
          "description": "True if the load balancer endpoint is public, false if it is
not. Default is false.",
          "type": "boolean",
          "default": false
        },
        "SSLCertificateId": {
```

```
          "description": "The Amazon Resource Name (ARN) of the
 SSL certificate to use, in the form arn:aws:acm:us-east-1:ACCOUNT-
ID:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "string"
        },
        "SubnetIds": {
          "description": "Two or more subnet IDs for the load balancer, in the form
 subnet-0123abcd or subnet-01234567890abcdef, spanning at least two Availability
 Zones.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 2,
          "uniqueItems": true
        },
        "ValidHTTPCode": {
          "default": "200",
          "description": "The HTTP codes that a healthy target application server must
 use when responding to a health check, such as 200, 202 or 200-399.",
          "type": "string",
          "pattern": "^[1-5][0-9]{2}(-[1-5][0-9]{2})?$"
        }
      },
      "additionalProperties": false,
      "required": [
        "SubnetIds"
      ]
    },
    "AutoScalingGroup": {
      "description": "Specifications for the ASG.",
      "type": "object",
      "properties": {
        "AmiId": {
          "description": "ID of the AMI for the Auto Scaling group to use when creating
 new instances, in the form ami-0123abcd or ami-01234567890abcdef.",
          "type": "string",
          "pattern": "^ami-[a-z0-9]{8}$|^ami-[a-z0-9]{17}$"
        },
        "CooldownInSeconds": {
          "description": "The number of seconds after a scaling activity is complete
 before any further scaling activities can start.",
          "type": "integer",
```

```
          "minimum": 120,
          "maximum": 600,
          "default": 300
        },
        "DesiredCapacity": {
          "description": "The number of EC2 instances you want running in the group.
This number must be greater than or equal to the MinInstances setting and less than or
equal to the MaxInstances setting.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 1
        },
        "EBSOptimized": {
          "description": "True to create EBS-optimized instances, false to not.
EBS-optimization provides dedicated throughput to Amazon EBS and optimal EBS I/O
performance.",
          "type": "boolean",
          "default": false
        },
        "HealthCheckGracePeriodInSeconds": {
          "description": "The amount of time, in seconds, that Auto Scaling waits
before checking the health status of an EC2 instance that has come into service.
During this time, any health check failures for the instance are ignored.",
          "type": "integer",
          "minimum": 600,
          "maximum": 1800,
          "default": 1800
        },
        "HealthCheckType": {
          "description": "The service to use for the health checks. The ELB Health
Check Type includes EC2 instance and system status checks. If ASGHealthCheckType =
ELB, ensure that your ASGHealthCheckGracePeriod value is long enough so that your
instances are not terminated due to load-balancer health checks failing, before your
application has been deployed.",
          "default": "EC2",
          "type": "string",
          "enum": [
            "EC2",
            "ELB"
          ]
        },
        "IAMInstanceProfile": {
```

```
        "description": "The IAM instance profile for the Auto Scaling group. EC2
instances launched with an IAM role automatically have AWS security credentials
available.",
        "type": "string",
        "default": "customer-mc-ec2-instance-profile"
      },
      "InstanceDetailedMonitoring": {
        "description": "True to enable detailed monitoring on the instances in the
Auto Scaling group, false to use only basic monitoring.",
        "type": "boolean",
        "default": true
      },
      "InstanceRootVolumeIops": {
        "description": "The IOPS to use for the root volume if io1 volume type is
specified.",
        "type": "integer",
        "minimum": 0,
        "maximum": 20000,
        "default": 0
      },
      "InstanceRootVolumeName": {
        "description": "The name of the root volume to use. Defaults to /dev/xvda for
Linux, and /dev/sda for Windows.",
        "type": "string"
      },
      "InstanceRootVolumeSizeInGiB": {
        "description": "The size of the root volume for the instance. Defaults to 20
GiB for Linux, and 60 GiB for Windows.",
        "type": "integer",
        "minimum": 8,
        "maximum": 1024
      },
      "InstanceRootVolumeType": {
        "description": "Choose io1 or gp2 for SSD-backed volumes optimized for
transactional workloads; choose standard for HDD-backed volumes optimized for large
streaming workloads.",
        "type": "string",
        "enum": [
          "standard",
          "io1",
          "gp2"
        ],
        "default": "standard"
      },
```

```
        "InstanceType": {
          "description": "The instance type for the Auto Scaling group to use when
creating new EC2 instances.",
          "type": "string",
          "default": "m4.large"
        },
        "MaxInstances": {
          "description": "The maximum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 1
        },
        "MinInstances": {
          "description": "The minimum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000,
          "default": 1
        },
        "ScaleMetricName": {
          "description": "The metric to use to in a scale-down event. Exceeding the
metric triggers an alarm.",
          "type": "string",
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
            "StatusCheckFailed_System"
          ],
          "default": "CPUUtilization"
        },
        "ScaleDownPolicyCooldownInSeconds": {
```

```
          "description": "The number of seconds after a scale-down activity is
completed before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600,
          "default": 300
        },
        "ScaleDownPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ScaleMetricName threshold.",
          "type": "integer",
          "minimum": 2,
          "default": 4
        },
        "ScaleDownPolicyPeriod": {
          "description": "The time over which the specified ScaleDownPolicyStatistic is
applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60,
          "default": 60
        },
        "ScaleDownPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale down.",
          "type": "integer",
          "maximum": 0,
          "default": -1
        },
        "ScaleDownPolicyStatistic": {
          "description": "The statistic to apply to the alarm's ScaleDownMetricName.",
          "type": "string",
          "enum": [
            "Average",
            "Maximum",
            "Minimum",
            "SampleCount",
            "Sum"
          ],
          "default": "Average"
        },
        "ScaleDownPolicyThreshold": {
          "description": "The value against which the specified
ScaleDownPolicyStatistic is compared.",
          "type": "number",
```

```
          "default": 35
        },
        "ScaleUpPolicyCooldownInSeconds": {
          "description": "The number of seconds after a scale-up activity is completed
before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600,
          "default": 300
        },
        "ScaleUpPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ScaleUpMetricName threshold.",
          "type": "integer",
          "minimum": 2,
          "default": 2
        },
        "ScaleUpPolicyPeriod": {
          "description": "The time over which the specified ScaleUpPolicyStatistic is
applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60,
          "default": 60
        },
        "ScaleUpPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale up.",
          "type": "integer",
          "minimum": 0,
          "default": 2
        },
        "ScaleUpPolicyStatistic": {
          "description": "The statistic to apply to the alarm's ScaleMetricName.",
          "type": "string",
          "enum": [
            "Average",
            "Maximum",
            "Minimum",
            "SampleCount",
            "Sum"
          ],
          "default": "Average"
        },
        "ScaleUpPolicyThreshold": {
```

```
          "description": "The value against which the specified ScaleUpPolicyStatistic
is compared.",
          "type": "number",
          "default": 75
        },
        "SubnetIds": {
          "description": "One or more subnets for the Auto Scaling group to launch
instances into (scale up) or remove instances from (scale down), in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 2,
          "uniqueItems": true
        },
        "UserData": {
          "description": "A comma-delimited list where each element is a line of script
to be run on boot.",
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 1,
          "default": [
            ""
          ]
        }
      },
      "additionalProperties": false,
      "required": [
        "AmiId",
        "SubnetIds"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "AutoScalingGroup",
    "ApplicationLoadBalancer",
    "Description",
    "Name",
```

```
    "TimeoutInMinutes",
    "VpcId"
  ]
}
```

Schema for Change Type ct-0ah3gwb9seqk2

Classifications:

- Deployment | Applications | CodeDeploy application | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create CodeDeploy application",
  "description": "Use to create an AWS CodeDeploy application resource with the
 specified name.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The reason for the request.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-sft6rv00000000000",
      "type": "string",
      "enum": [
        "stm-sft6rv00000000000"
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
```

```
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "CodeDeployApplicationName": {
          "description": "The name of an AWS CodeDeploy application.",
          "type": "string",
          "minLength": 1,
```

```
          "maxLength": 100,
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        }
      },
      "additionalProperties": false,
      "required": [
        "CodeDeployApplicationName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0aqx5t0pgfzbg

Classifications:

- Management | Advanced stack components | Load balancer (ELB) stack | Replace listener certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Replace ELB Listener Certificate",
  "description": "Replace the certificate of an existing Elastic (Classic) Load
 Balancer (ELB) listener. Use the RemediateDrift parameter to have the automation try
 to remediate the stack drift, if drift is introduced in the CloudFormation stack that
 was used to create the load balancer.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-SetClassicLoadBalancerCertificate.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-SetClassicLoadBalancerCertificate"
      ],
      "default": "AWSManagedServices-SetClassicLoadBalancerCertificate"
    },
    "Region": {
      "description": "The AWS Region where the ELB listener is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "LoadBalancerName": {
          "description": "The name of the Classic Load Balancer.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-]{1,30}[a-zA-Z0-9]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SSLCertificateArn": {
          "description": "The Amazon Resource Name (ARN) of the certificate in the form
 arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "LoadBalancerPort": {
          "description": "The listener port of the Classic Load Balancer.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{2,5}$"
          },
          "minItems": 1,
```

```
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused by
replacing the certificate on the Load Balancer listener. False to not attempt drift
remediation. Drift remediation can be performed only on CloudFormation stacks that
were created using a CT other than the Ingestion CT ct-36cn2avfrrj9v and that are in
sync with the definitions in the stack template prior to setting certificate to the
Load Balancer listener. Set to False to replace the certificate on the Load Balancer
listener in an ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "LoadBalancerName",
          "SSLCertificateArn",
          "LoadBalancerPort",
          "RemediateStackDrift"
        ]
      },
      "additionalProperties": false,
      "required": [
        "LoadBalancerName",
        "SSLCertificateArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
```

```
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0ary07xiajwx4

Classifications:

- Deployment | Advanced stack components | Load balancer (ELB) stack | Create (with additional listeners)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Load Balancer (ELB)",
  "description": "Create an Elastic (\"Classic\") load balancer (ELB).",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name used in the Console.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
```

```
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-3tdleig07sbhstgnf",
      "type": "string",
      "enum": [
        "stm-3tdleig07sbhstgnf"
      ],
      "default": "stm-3tdleig07sbhstgnf"
    },
    "TimeoutInMinutes": {
```

```
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "LoadBalancer": {
      "type": "object",
      "properties": {
        "Name": {
          "type": "string",
          "description": "A friendly name for the load balancer.",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,31}$|^$"
        },
        "Scheme": {
          "type": "string",
          "description": "True if the load balancer endpoint is public, false if it is
private.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "SecurityGroups": {
          "type": "array",
          "description": "A list of security groups to associate with the load
balancer.",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 5,
          "uniqueItems": true
        },
        "SubnetIds": {
          "type": "array",
          "description": "A list of subnet IDs that the Elastic Load Balancing creates
load balancer nodes in. For an Internet-facing load balancer provide a public subnet
ID, for an internal load balancer we recommend private subnet IDs.",
          "items": {
```

```
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "uniqueItems": true
        },
        "AccessLogInterval": {
          "type": "string",
          "description": "The time interval, in minutes, to upload the load balancer
access log to the specified S3 bucket. Defaults to 60 Minutes.",
          "enum": [
            "5",
            "60"
          ],
          "default": "60"
        },
        "ConnectionDrainingTimeout": {
          "type": "integer",
          "description": "The maximum time, in seconds, to keep the existing
connections open before deregistering the instances.",
          "default": 60,
          "minimum": 1,
          "maximum": 3600
        },
        "IdleTimeout": {
          "type": "integer",
          "description": "The time, in seconds, that a connection to the load balancer
can remain idle (no data is sent over the connection). After the specified time, the
load balancer closes the connection.",
          "default": 60,
          "minimum": 1,
          "maximum": 3600
        },
        "CrossZone": {
          "type": "string",
          "description": "True to enable cross-zone load balancing (the load balancer
nodes route traffic to the back-end instances across all Availability Zones), false to
disable. Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "HealthCheckHealthyThreshold": {
```

```
          "type": "string",
          "description": "The number of consecutive health probe successes required
 before moving the instance to the healthy state after it was moved to unhealthy.",
          "pattern": "[1-9]{1}[0-9]{0,1}",
          "default": "2"
        },
        "HealthCheckInterval": {
          "type": "string",
          "description": "How often, in seconds, that health checks are run on an
 individual load balancer node.",
          "pattern": "[1-9]{1}[0-9]{0,3}",
          "default": "10"
        },
        "HealthCheckTarget": {
          "type": "string",
          "description": "The protocol, port, and path of the instance to check. The
 protocol can be TCP, HTTP, HTTPS, or SSL and valid ports are 1 through 65535. For TCP/
SSL no path is required. For HTTP/HTTPS, you must include a ping path in the string.
 For example, HTTP:80/weather/us/wa/seattle.",
          "pattern": "(HTTP|HTTPS):[0-9]{1,5}[/][\\w./-]*|(SSL|TCP):[0-9]{1,5}",
          "default": "TCP:80"
        },
        "HealthCheckTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, during which no
 response means a failed health probe. This value must be less than the value for
 HealthCheckInterval.",
          "pattern": "[1-9]{1}[0-9]{0,3}",
          "default": "5"
        },
        "HealthCheckUnhealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health probe failures required
 before moving the instance to the unhealthy state.",
          "pattern": "[1-9]{1}[0-9]{0,2}",
          "default": "10"
        },
        "BackendInstances": {
          "type": "array",
          "description": "A list of EC2 instance IDs to associate with the load
 balancer, in the form of i-0123abcd or i-01234567890abcdef for a single instance,
 or i-0123abcd,i-12345abcd or i-01234567890abcdef,i-2345678901abcdefg for multiple
 instances. Leave blank to not associate individual EC2 instances with the load
 balancer. A load balancer can be associated with an autoscaling group by specifying
```

```
the load balancer name in the ASGLoadBalancerNames property during creation or update
of the autoscaling group.",
          "items": {
            "type": "string",
            "pattern": "^i-([0-9a-zA-Z]{8}|[0-9a-zA-Z]{17})$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "LBCookieExpirationPeriod": {
          "type": "string",
          "description": "The time period, in seconds, after which the cookie is
considered stale. If this parameter isn't specified, the sticky session will last for
the duration of the browser session.",
          "pattern": "^[0-9]+$|^$"
        },
        "LBCookieStickinessPolicyName": {
          "type": "string",
          "description": "A name for the load balancer cookie stickiness policy. The
name must be unique within the set of policies for this load balancer. To associate
with a listener, specify the name under PolicyNames in the respective listener
configuration.",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "AppCookieName": {
          "type": "string",
          "description": "A name for the application cookie used for stickiness.",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "AppCookiePolicyName": {
          "type": "string",
          "description": "A name for the application cookie stickiness policy. The
name must be unique within the set of policies for this load balancer. To associate
with a listener, specify the name under PolicyNames in the respective listener
configuration.",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "Scheme",
          "SecurityGroups",
          "SubnetIds",
```

```
          "BackendInstances",
          "IdleTimeout",
          "CrossZone",
          "AccessLogInterval",
          "ConnectionDrainingTimeout",
          "HealthCheckHealthyThreshold",
          "HealthCheckInterval",
          "HealthCheckTarget",
          "HealthCheckTimeout",
          "HealthCheckUnhealthyThreshold",
          "LBCookieExpirationPeriod",
          "LBCookieStickinessPolicyName",
          "AppCookieName",
          "AppCookiePolicyName"
        ]
      },
      "required": [
        "SecurityGroups",
        "SubnetIds"
      ],
      "additionalProperties": false
    },
    "Listener1": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$",
          "default": "80"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ],
          "default": "HTTP"
        },
```

```
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$",
          "default": "80"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ],
          "default": "HTTP"
        },
        "PolicyNames": {
          "type": "array",
          "description": "A list of policy names to associate with the listener.",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
```

```
          "InstanceProtocol",
          "PolicyNames",
          "SSLCertificateId"
        ]
      },
      "required": [
        "Port",
        "Protocol",
        "InstancePort"
      ],
      "additionalProperties": false
    },
    "Listener2": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
```

```
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "PolicyNames": {
          "type": "array",
          "description": "A list of policy names to associate with the listener.",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "PolicyNames",
          "SSLCertificateId"
        ]
      },
      "additionalProperties": false
    },
    "Listener3": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
```

```
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "PolicyNames": {
          "type": "array",
          "description": "A list of policy names to associate with the listener.",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SSLCertificateId": {
          "type": "string",
```

```
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "PolicyNames",
          "SSLCertificateId"
        ]
      },
      "additionalProperties": false
    },
    "Listener4": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
```

```
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "PolicyNames": {
          "type": "array",
          "description": "A list of policy names to associate with the listener.",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "PolicyNames",
          "SSLCertificateId"
        ]
      },
      "additionalProperties": false
```

```
    },
    "Listener5": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "PolicyNames": {
          "type": "array",
          "description": "A list of policy names to associate with the listener.",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
```

```
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "PolicyNames",
          "SSLCertificateId"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags",
      "LoadBalancer",
      "Listener1",
      "Listener2",
      "Listener3",
      "Listener4",
      "Listener5"
    ]
  },
  "required": [
```

```
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "StackTemplateId",
    "LoadBalancer",
    "Listener1"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0attesnjqy2cx

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS source endpoint",
  "description": "Use to create a Database Migration Service (DMS) source endpoint.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
```

```
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-pud4ghhkp7395n9bc.",
      "type": "string",
      "enum": [
        "stm-pud4ghhkp7395n9bc"
      ],
      "default": "stm-pud4ghhkp7395n9bc"
```

```
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) for the certificate to use
with the source. This is required if SslMode = verify-ca or verify-full.",
          "pattern": "^$|^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:cert:[A-Z0-9]+$"
        },
        "DatabaseName": {
          "type": "string",
          "description": "The name of the source database. Must not be blank if
EngineName = azuredb, db2, oracle, postgres, sqlserver or sybase."
        },
        "EndpointIdentifier": {
          "type": "string",
          "description": "A meaningful identifier for the source database endpoint.
Must be unique for all endpoints owned by your AWS account in the current region. Must
begin with a letter, must contain only ASCII letters, digits and hyphens and must not
end with a hyphen or contain two consecutive hyphens.",
          "pattern": "^$|(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$"
        },
        "EngineName": {
          "type": "string",
          "description": "The type of engine this source endpoint is connected to. Some
parameters become required depending on the specified EngineName.",
          "enum": [
            "aurora",
            "azuredb",
            "db2",
            "mariadb",
            "mysql",
            "oracle",
            "postgres",
```

```
            "sqlserver",
            "sybase"
          ]
        },
        "ExtraConnectionAttributes": {
          "type": "string",
          "description": "Additional attributes associated with the connection. See AWS
documentation for more information on the supported extra connection attributes for
the EngineName you have selected."
        },
        "KmsKeyId": {
          "type": "string",
          "description": "The AWS Key Management Service (AWS KMS) customer master key
(CMK) ID to use for encrypting volumes associated with the replication instance. If
not specified, the default CMK for Amazon DMS is used.",
          "pattern": "^$|^[\\w]{8}-[\\w]{4}-[\\w]{4}-[\\w]{4}-[\\w]{12}$"
        },
        "Password": {
          "type": "string",
          "description": "The password to be used to log in to the source database.",
          "metadata": {
            "ams:sensitive": true
          }
        },
        "Port": {
          "type": "integer",
          "description": "The port used by the source database.",
          "minimum": 1,
          "maximum": 65535
        },
        "ServerName": {
          "type": "string",
          "description": "The name of the server where the source database resides."
        },
        "SslMode": {
          "type": "string",
          "description": "The SSL mode to use for the SSL connection.",
          "enum": [
            "none",
            "require",
            "verify-ca",
            "verify-full"
          ],
          "default": "none"
```

```
        },
        "Username": {
          "type": "string",
          "description": "The user name to be used to log in to the source database.",
          "metadata": {
            "ams:sensitive": true
          }
        }
      },
      "metadata": {
        "ui:order": [
          "EndpointIdentifier",
          "EngineName",
          "ServerName",
          "Port",
          "DatabaseName",
          "Username",
          "Password",
          "SslMode",
          "CertificateArn",
          "KmsKeyId",
          "ExtraConnectionAttributes"
        ]
      },
      "required": [
        "EngineName",
        "ServerName",
        "Port",
        "Username",
        "Password"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
```

```
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0bpxsrtu16igp

Classifications:

- Management | Advanced stack components | RDS database stack | Reboot

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Reboot RDS DB instance",
  "description": "Use to reboot an RDS DB instance.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "DbInstanceIdentifier": {
      "pattern": "(?=[a-zA-Z0-9-]{1,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$",
      "description": "The identifier of the DB instance to reboot.",
      "type": "string"
    },
    "ForceFailover": {
      "default": false,
      "description": "True to reboot with Multi-AZ failover, for Multi-AZ instances.
 Default is false.",
      "type": "boolean"
    }
  },
  "required": [
    "DbInstanceIdentifier"
  ]
}
```

Schema for Change Type ct-0c38gftq56zj6

Classifications:

- Deployment | Advanced stack components | DNS (private) | Create


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Private DNS Record",
  "description": "Create a new Route 53 DNS resource record sets and a new private
 hosted zone for a VPC, and configure traffic routing.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateAddRoute53Resources.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateAddRoute53Resources"
      ],
      "default": "AWSManagedServices-CreateAddRoute53Resources"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "DomainName": {
          "description": "A domain name for the hosted zone. The name can contain only
 lowercase letters, numbers, hyphens (-), and a dot (.). For example, mycorp.com",
          "type": "string",
          "minLength": 2,
          "pattern": "^([a-z0-9]+(-[a-z0-9]+)*\\.)+[a-z]{2,255}$"
        },
        "VPCId": {
          "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
          "type": "string",
```

```
          "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
        },
        "DomainType": {
          "description": "Must be 'private'",
          "type": "string",
          "enum": [
            "private"
          ],
          "default": "private"
        },
        "RecordSet": {
          "description": "A JSON of resource records for the hosted zone.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\s*\\{\\s*\"RecordSet\"\\s*:\\s*\\[.*\\]\\s*\\}\\s*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "DomainName",
          "VPCId",
          "DomainType",
          "RecordSet"
        ]
      },
      "required": [
        "DomainName",
        "VPCId",
        "DomainType",
        "RecordSet"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
```

```
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0cupn1txog5tk

Classifications:

- Deployment | Advanced stack components | Storage Gateway | Create from Backup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start Storage Gateway Restore Job",
  "description": "Start an AWS Backup service restore job to restore a Storage Gateway
 volume snapshot of the specified resource.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
StartRestoreJobStorageGatewayVolume.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartRestoreJobStorageGatewayVolume"
      ],
      "default": "AWSManagedServices-StartRestoreJobStorageGatewayVolume"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RecoveryPointArn": {
```

```
          "description": "The Amazon Resource Name (ARN) that uniquely identifies a
 recovery point.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "BackupVaultName": {
          "description": "The name of the target backup vault. The backup vault name is
 case sensitive and must contain from 2 to 50 alphanumeric characters or hyphens.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "GatewayArn": {
          "description": "The Amazon Resource Name (ARN) that uniquely identifies a
 Storage Gateway.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TargetName": {
          "description": "The name of the Internet Small Computer Systems
 Interface(iSCSI) target. This is the name your iSCSI initiator uses to connect to
 your volume. The target name can contain lowercase letters, numbers, periods (.), and
 hyphens (-).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9\\_\\-\\.]+$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        },
        "GatewayType": {
          "description": "The Storage Gateway volume restore type. For data that is
 cached in the gateway and stored in S3, choose Cached. For on-premise data stored
 locally, choose Stored. If you choose Stored, you must also specify a DiskId.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Cached",
              "Stored"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DiskId": {
          "description": "The unique identifier for the gateway local disk that is
 configured as a stored volume. Find disk IDs for a gateway on the Storage Gateway
 console. Required when GatewayType = Stored. If specified, all data currently residing
 on this disk will be lost, and overwritten with the current data on the snapshot.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": "^(|[a-z0-9\\_\\-\\.\\:]+)$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "VolumeSize": {
          "description": "The size of the volume, in GiBs. If this value is specified,
 it must be greater than the snapshot size, to take affect. By default, the volume size
 is equal to the snapshot size.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "0",
            "pattern": "^(0|[1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|1[0-5][0-9]{3}|16[0-2][0-9]{2}|
163[0-7][0-9]|1638[0-4])$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        },
        "IamRoleArn": {
          "description": "The ARN of the role that allows AWS Backup to perform the
 actions on your behalf. If no role is specified, the default IAM role, created by AMS
 during the account onboarding process, is used.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": "^(|arn:aws:iam:([a-z]{2}((-gov))?-[a-z]+-[0-9]){0,1}:[0-9]
{12}:role\\/[a-zA-Z0-9\\_\\-]+)$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KmsKeyArn": {
          "description": "The Amazon Resource Name (ARN) for the AWS KMS key to encrypt
 the new Storage Gateway volume.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": "^(|arn:aws:kms:([a-z]{2}((-gov))?-[a-z]+-\\d{1}):[0-9]{0,12}:
[a-zA-Z0-9\\_\\-\\/\\:]+)$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "RecoveryPointArn",
          "BackupVaultName",
          "GatewayArn",
          "TargetName",
          "GatewayType",
          "DiskId",
          "VolumeSize",
          "IamRoleArn",
          "KmsKeyArn"
        ]
      },
```

```
      "required": [
        "RecoveryPointArn",
        "BackupVaultName",
        "GatewayArn",
        "TargetName",
        "GatewayType"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0cyqd7laxyhlm

Classifications:

- Deployment | Monitoring and notification | CloudWatch | Create LogGroup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "CloudWatch LogGroup with optional subscription filter, log streams and
 metric filters.",
  "description": "Creates a CloudWatch LogGroup with optional subscription filter, up
 to 5 log streams and up to 5 metric filters.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
```

```
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
```

```
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-8ian3plt5a6jbv7jt",
      "type": "string",
      "enum": [
        "stm-8ian3plt5a6jbv7jt"
      ],
      "default": "stm-8ian3plt5a6jbv7jt"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "LogGroupName": {
          "type": "string",
          "description": "A name for the log group. The name must be prefixed with the
word 'customer'.",
          "pattern": "^customer[a-zA-Z0-9\\.\\-_/#]{1,504}$"
        },
        "LogGroupRetentionInDays": {
          "type": "string",
          "description": "The number of days to retain the log events in the log group
created. Leave blank to keep logs indefinitely.",
          "enum": [
            "",
            "1",
            "3",
            "5",
            "7",
            "14",
```

```
            "30",
            "60",
            "90",
            "120",
            "150",
            "180",
            "365",
            "400",
            "545",
            "731",
            "1827",
            "3653"
          ],
          "default": ""
        },
        "LogStream1Name": {
          "type": "string",
          "description": "A name for log stream 1. The name must be unique within the
log group. If left blank log stream 1 is not created.",
          "pattern": "^[a-zA-Z0-9\\.\\-_/#]{1,512}$|^$",
          "default": ""
        },
        "LogStream2Name": {
          "type": "string",
          "description": "A name for log stream 2. The name must be unique within the
log group. If left blank log stream 2 is not created.",
          "pattern": "^[a-zA-Z0-9\\.\\-_/#]{1,512}$|^$",
          "default": ""
        },
        "LogStream3Name": {
          "type": "string",
          "description": "A name for log stream 3. The name must be unique within the
log group. If left blank log stream 3 is not created.",
          "pattern": "^[a-zA-Z0-9\\.\\-_/#]{1,512}$|^$",
          "default": ""
        },
        "LogStream4Name": {
          "type": "string",
          "description": "A name for log stream 4. The name must be unique within the
log group. If left blank log stream 4 is not created.",
          "pattern": "^[a-zA-Z0-9\\.\\-_/#]{1,512}$|^$",
          "default": ""
        },
        "LogStream5Name": {
```

```
          "type": "string",
          "description": "A name for log stream 5. The name must be unique within the
 log group. If left blank log stream 5 is not created.",
          "pattern": "^[a-zA-Z0-9\\.\\-_/#]{1,512}$|^$",
          "default": ""
        },
        "SubscriptionFilterIAMroleARN": {
          "type": "string",
          "description": "An IAM role that grants CloudWatch Logs permission to put
 data into the destination. Applicable only if the destination is Kinesis stream or
 Kinesis Data Firehose delivery stream.",
          "pattern": "(arn:aws:iam::\\d{12}:role\\/[\\w+=,.@-]{1,64}|^$)",
          "default": ""
        },
        "SubscriptionFilterPattern": {
          "type": "string",
          "description": "The filtering expressions that restrict what gets delivered
 to the destination AWS resource.",
          "pattern": "^.{1,1024}$|^$",
          "default": ""
        },
        "SubscriptionDestinationARN": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the Kinesis stream, Kinesis
 Data Firehose delivery stream, or Lambda function, to use as the subscription feed
 destination.",
          "pattern": "^arn:aws:kinesis:[a-z0-9-]+:[0-9]{12}:stream/[a-zA-Z0-9-_\\.]
{1,128}$|^arn:aws:firehose:[a-z0-9-]+:[0-9]{12}:deliverystream/[a-zA-Z0-9-_\\.]{1,64}$|
^arn:aws:lambda:[a-z0-9-]+:[0-9]{12}:function:[a-zA-Z0-9-_]{1,140}$|^$",
          "default": ""
        },
        "MetricFilter1Pattern": {
          "type": "string",
          "description": "The pattern for MetricFilter1 that CloudWatch Logs follows to
 interpret each entry in a log.",
          "pattern": "^.{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter1DefaultValue": {
          "type": "string",
          "description": "The value to emit when a filter pattern does not match a log
 event.",
          "pattern": "^[0-9]{1,100}$|^$",
          "default": ""
```

```
      },
      "MetricFilter1Value": {
        "type": "string",
        "description": "The value that is published to the CloudWatch metric. If left
blank MetricFilter1 is not created.",
        "pattern": "^[0-9]{1,100}$|^$",
        "default": ""
      },
      "MetricFilter1Namespace": {
        "type": "string",
        "description": "The destination namespace of the CloudWatch metric for the
MetricFilter1. Namespaces are containers for metrics. If left blank MetricFilter1 is
not created.",
        "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter1Name": {
        "type": "string",
        "description": "The name of the CloudWatch metric that the log information is
published to. If left blank MetricFilter1 is not created.",
        "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter2Pattern": {
        "type": "string",
        "description": "The pattern for MetricFilter2 that CloudWatch Logs follows to
interpret each entry in a log.",
        "pattern": "^.{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter2DefaultValue": {
        "type": "string",
        "description": "The value to emit when a filter pattern does not match a log
event.",
        "pattern": "^[0-9]{1,100}$|^$",
        "default": ""
      },
      "MetricFilter2Value": {
        "type": "string",
        "description": "The value that is published to the CloudWatch metric. If left
blank MetricFilter2 is not created.",
        "pattern": "^[0-9]{1,100}$|^$",
        "default": ""
      },
```

```
      "MetricFilter2Namespace": {
        "type": "string",
        "description": "The destination namespace of the CloudWatch metric for the
MetricFilter2. Namespaces are containers for metrics. If left blank MetricFilter2 is
not created.",
        "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter2Name": {
        "type": "string",
        "description": "The name of the CloudWatch metric that the log information is
published to. If left blank MetricFilter2 is not created.",
        "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter3Pattern": {
        "type": "string",
        "description": "The pattern for MetricFilter3 that CloudWatch Logs follows to
interpret each entry in a log.",
        "pattern": "^.{1,1024}$|^$",
        "default": ""
      },
      "MetricFilter3DefaultValue": {
        "type": "string",
        "description": "The value to emit when a filter pattern does not match a log
event.",
        "pattern": "^[0-9]{1,100}$|^$",
        "default": ""
      },
      "MetricFilter3Value": {
        "type": "string",
        "description": "The value that is published to the CloudWatch metric. If left
blank MetricFilter3 is not created.",
        "pattern": "^[0-9]{1,100}$|^$",
        "default": ""
      },
      "MetricFilter3Namespace": {
        "type": "string",
        "description": "The destination namespace of the CloudWatch metric for the
MetricFilter3. Namespaces are containers for metrics. If left blank MetricFilter3 is
not created.",
        "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
        "default": ""
      },
```

```
        "MetricFilter3Name": {
          "type": "string",
          "description": "The name of the CloudWatch metric that the log information is
published to. If left blank MetricFilter3 is not created.",
          "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter4Pattern": {
          "type": "string",
          "description": "The pattern for MetricFilter4 that CloudWatch Logs follows to
interpret each entry in a log.",
          "pattern": "^.{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter4DefaultValue": {
          "type": "string",
          "description": "The value to emit when a filter pattern does not match a log
event.",
          "pattern": "^[0-9]{1,100}$|^$",
          "default": ""
        },
        "MetricFilter4Value": {
          "type": "string",
          "description": "The value that is published to the CloudWatch metric. If left
blank MetricFilter4 is not created.",
          "pattern": "^[0-9]{1,100}$|^$",
          "default": ""
        },
        "MetricFilter4Namespace": {
          "type": "string",
          "description": "The destination namespace of the CloudWatch metric for the
MetricFilter4. Namespaces are containers for metrics. If left blank MetricFilter4 is
not created.",
          "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter4Name": {
          "type": "string",
          "description": "The name of the CloudWatch metric that the log information is
published to. If left blank MetricFilter4 is not created.",
          "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter5Pattern": {
```

```
          "type": "string",
          "description": "The pattern for MetricFilter5 that CloudWatch Logs follows to
interpret each entry in a log.",
          "pattern": "^.{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter5DefaultValue": {
          "type": "string",
          "description": "The value to emit when a filter pattern does not match a log
event.",
          "pattern": "^[0-9]{1,100}$|^$",
          "default": ""
        },
        "MetricFilter5Value": {
          "type": "string",
          "description": "The value that is published to the CloudWatch metric. If left
blank MetricFilter5 is not created.",
          "pattern": "^[0-9]{1,100}$|^$",
          "default": ""
        },
        "MetricFilter5Namespace": {
          "type": "string",
          "description": "The destination namespace of the CloudWatch metric for the
MetricFilter5. Namespaces are containers for metrics. If left blank MetricFilter5 is
not created.",
          "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
          "default": ""
        },
        "MetricFilter5Name": {
          "type": "string",
          "description": "The name of the CloudWatch metric that the log information is
published to. If left blank MetricFilter5 is not created.",
          "pattern": "^[a-zA-Z0-9_\\-\\/.]{1,1024}$|^$",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "LogGroupName",
          "LogGroupRetentionInDays",
          "LogStream1Name",
          "LogStream2Name",
          "LogStream3Name",
          "LogStream4Name",
```

```
          "LogStream5Name",
          "SubscriptionFilterIAMroleARN",
          "SubscriptionFilterPattern",
          "SubscriptionDestinationARN",
          "MetricFilter1Name",
          "MetricFilter1Namespace",
          "MetricFilter1Pattern",
          "MetricFilter1Value",
          "MetricFilter1DefaultValue",
          "MetricFilter2Name",
          "MetricFilter2Namespace",
          "MetricFilter2Pattern",
          "MetricFilter2Value",
          "MetricFilter2DefaultValue",
          "MetricFilter3Name",
          "MetricFilter3Namespace",
          "MetricFilter3Pattern",
          "MetricFilter3Value",
          "MetricFilter3DefaultValue",
          "MetricFilter4Name",
          "MetricFilter4Namespace",
          "MetricFilter4Pattern",
          "MetricFilter4Value",
          "MetricFilter4DefaultValue",
          "MetricFilter5Name",
          "MetricFilter5Namespace",
          "MetricFilter5Pattern",
          "MetricFilter5Value",
          "MetricFilter5DefaultValue"
        ]
      },
      "required": [
        "LogGroupName"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
```

```
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0el2j07llrxs7

Classifications:

- Deployment | Patching | SSM patch window | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Window",
  "description": "Create an AWS Systems Manager (SSM) patch window for patching to take
 place on instances with the specified PatchGroup. The patch window is an SSM resource
 that you can manage with the SSM console.",
  "properties": {
    "Cutoff": {
      "description": "The maximum number of hours before the end of the scheduled patch
 window for starting a new patching command. This helps ensure that patching commands
 complete before the patch window ends. A new patching command can only start execution
 within the patch window and before the specified Cutoff. After the Cutoff is reached,
 no new patching commands can be started.",
      "default": 0,
      "maximum": 23,
      "minimum": 0,
      "type": "integer"
    },
    "Description": {
      "description": "A meaningful description for this patch window.",
      "maxLength": 500,
```

```
      "minLength": 1,
      "type": "string"
    },
    "Duration": {
      "description": "The duration of the patch window in hours.",
      "maximum": 24,
      "minimum": 1,
      "type": "integer"
    },
    "EndDate": {
      "description": "The date and time, in ISO-8601 extended format, for when the
patch window is scheduled to become inactive (i.e.: 2019-10-23T19:45:00Z).",
      "type": "string"
    },
    "MaxConcurrency": {
      "description": "The maximum number or rate (%) of instances allowed to patch in
parallel.",
      "default": "33%",
      "maxLength": 7,
      "minLength": 1,
      "pattern": "^([1-9][0-9]*|[1-9][0-9]%|[1-9]%|100%)$",
      "type": "string"
    },
    "MaxErrors": {
      "description": "The maximum number or rate (%) of errors allowed before the
Patching stops being scheduled.",
      "default": "100%",
      "maxLength": 7,
      "minLength": 1,
      "pattern": "^([1-9][0-9]*|[1-9][0-9]%|[1-9]%|100%)$",
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch window.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "NotificationEmails": {
      "description": "One or more email addresses to receive notifications about
patching status.",
      "type": "array",
      "items": {
```

```
        "type": "string",
        "pattern": "^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$"
      },
      "minItems": 1,
      "maxItems": 5,
      "uniqueItems": true
    },
    "PatchGroup": {
      "description": "The value of the \"Patch Group\" tag of an existing instance;
 for example 'App123-CustA-EnvTest'. Instances with the specified \"Patch Group\" tag
 values, are included in the patch window. If needed, you can create \"Patch Group
\" tags using the console for the resource, but these tags are usually created at
 onboarding.",
      "type": "string",
      "minLength": 1,
      "maxLength": 256
    },
    "Schedule": {
      "description": "The schedule of the patch window in the form of a cron or rate
 expression; for example, cron(30 09 ? * * *) or rate(7 days).",
      "maxLength": 256,
      "minLength": 1,
      "type": "string"
    },
    "ScheduleOffset": {
      "description": "The number of days to wait after the date and time specified by a
 cron expression before the maintenance window runs.",
      "default": 0,
      "maximum": 6,
      "minimum": 0,
      "type": "integer"
    },
    "ScheduleTimeZone": {
      "description": "The time zone that the scheduled patch window executions are
 based on, in Internet Assigned Numbers Authority (IANA) format (i.e.: UTC, America/
Los_Angeles).",
      "default": "UTC",
      "pattern": "^[a-zA-Z_]+(\\+|/)?[a-zA-Z0-9_-]*(\\+|/)?[a-zA-Z0-9_-]+$",
      "type": "string"
    },
    "StartDate": {
      "description": "The date and time, in ISO-8601 extended format, after which the
 patch window becomes active (i.e.: 2019-10-23T19:45:00Z).",
      "type": "string"
```

```
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "PatchGroup",
      "Schedule",
      "ScheduleOffset",
      "Duration",
      "MaxConcurrency",
      "MaxErrors",
      "Cutoff",
      "StartDate",
      "EndDate",
      "ScheduleTimeZone",
      "NotificationEmails"
    ]
  },
  "required": [
    "Cutoff",
    "Duration",
    "MaxConcurrency",
    "MaxErrors",
    "Name",
    "NotificationEmails",
    "PatchGroup",
    "Schedule",
    "ScheduleTimeZone"
  ],
  "type": "object"
}
```

Schema for Change Type ct-0erkoad6uyvvg

Classifications:

- Management | Monitoring and notification | CloudWatch | Enable Non-Root Volumes Monitoring

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
  "name": "Enable Non-Root Volumes Monitoring",
  "description": "Enable monitoring on non-root volumes of an EC2 instance.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeployNonRootVolumeMonitoring.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeployNonRootVolumeMonitoring"
      ],
      "default": "AWSManagedServices-DeployNonRootVolumeMonitoring"
    },
    "Region": {
      "description": "The AWS Region where the EC2 instance, and volumes, are.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance, in the form i-12345678 or
i-123456789012345ab.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[0-9a-f]{8}$|^i-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId"
      ]
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0ffvihqwjvqj1

Classifications:

- Management | Advanced stack components | EC2 instance stack | Restore volumes

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Restore EC2 Volumes From Backup",
  "description": "Replace the instance volumes from an existing backup image of the
 instance.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
ReplaceInstanceVolumesFromSnapshotsWithContext. To restore from snapshot, use version 1
 of this change type.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ReplaceInstanceVolumesFromSnapshotsWithContext"
      ],
      "default": "AWSManagedServices-ReplaceInstanceVolumesFromSnapshotsWithContext"
    },
    "Region": {
      "description": "The AWS Region in which the EC2 instance is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
```

```
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The identifier of the EC2 instance to replace the volumes
 from the backup.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8,17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Backup": {
          "description": "The Amazon EC2 backup ARN, or AMI ID, custom or from backup,
 to use to restore the volumes, i.e. ami-0ecdf967356c809c7.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:ec2:[\\w]{2}-[a-z]+-[0-9]{1}::image/[A-Za-z0-9_-]+$|
^ami-[a-z0-9]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KMSKeyId": {
          "description": "The KMS key identifier, or ARN, to encrypt all restored
 volumes on the EC2 instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12}$|
^arn:aws:kms:[a-z]{2}-[a-z]+-\\d{1}:[0-9]{12}:key/[a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-
[a-z0-9]{4}-[a-z0-9]{12}$|^$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SleepTime": {
          "description": "The sleep time (how long to wait) before attempting access
 validation after data restoration completes.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^PT([0-9]|[1-5][0-9]|60)M$",
            "default": "PT5M"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ChangeHostname": {
          "description": "True to change the hostname after the restore operation, to a
 generated hostname. False to not change the hostname. Default is False.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "True",
              "False"
            ],
            "default": "False"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "Backup",
          "KMSKeyId",
          "ChangeHostname",
          "SleepTime"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId",
        "Backup"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
```

```
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0fpjlxa808sh2

Classifications:

- Management | Advanced stack components | S3 storage | Update policy (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update policy",
  "description": "Update an S3 bucket policy.",
  "type": "object",
  "properties": {
    "BucketName": {
      "description": "The name of the Amazon S3 bucket to which the policy applies.",
      "type": "string",
      "pattern": "^(?!(mc|ams|awsms)-)[a-z0-9][-.a-z0-9]{1,61}[a-z0-9]$"
    },
    "BucketPolicy": {
      "description": "Detailed information about the bucket permissions update, or
 a policy document to be attached to the bucket (paste the policy document into the
 value field). Details should include the type of access (for example Read, Write or
 Delete).",
      "type": "string",
      "maxLength": 20000
    },
    "PolicyAction": {
      "description": "Whether the given bucket policy needs to be appended to the
 existing bucket policy or to replace the bucket policy entirely. If you want to add
 a new statement block to the existing policy, choose 'Append'. If you want to replace
 the entire policy or update the policy in specific sections, provide the entire policy
 containing desired changes and choose 'Replace'.",
```

```
      "type": "string",
      "enum": [
        "Append",
        "Replace"
      ]
    },
    "Operation": {
      "description": "Must be Update policy.",
      "type": "string",
      "default": "Update policy",
      "enum": [
        "Update policy"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "BucketName",
      "BucketPolicy",
      "PolicyAction",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "BucketName",
    "BucketPolicy",
    "PolicyAction",
    "Operation"
  ]
}
```

Schema for Change Type ct-0fqo03yizfnw6

Classifications:

- Management | AWS Backup | Backup plan | Enable cross region copy


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Cross Region Copy",
  "description": "Update an existing backup plan rule with copy actions like cross
 region destination vault, and storage retention settings.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ConfigureCrossRegionBackup.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ConfigureCrossRegionBackup"
      ],
      "default": "AWSManagedServices-ConfigureCrossRegionBackup"
    },
    "Region": {
      "description": "The AWS Region in which the AWS Backup plan is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BackupPlanName": {
          "description": "The name of the existing Backup plan to be updated.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "maxItems": 1
        },
        "RuleName": {
          "description": "The name of the existing rule in the specified backup plan to
 be updated.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "maxItems": 1
        },
        "DestinationRegion": {
          "description": "The AWS Region where the destination backup vault is.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
          },
          "maxItems": 1
        },
        "DestinationVaultName": {
          "description": "The destination backup vault for the copied backup. If the
 vault does not exist in the destination Region, it is created automatically.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$",
            "default": "ams-replication-vault"
          },
          "maxItems": 1
        },
        "DestinationEncryptionKeyArn": {
          "description": "The destination server-side encryption key that is used to
 protect your backups. If the vault name does not exist and you do not provide a key
 ARN, a new key is created in the destination Region. For disaster recovery patterns,
 we recommend that you provide a key that belongs to a different account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(|arn:aws:kms:([a-z]{2}((-gov))?-[a-z]+-\\d{1}):[0-9]{0,12}:
[a-zA-Z0-9\\_\\-\\/\\:]+)$",
            "default": ""
          },
          "maxItems": 1
        },
        "DeleteAfterNumberOfDays": {
          "description": "The number of days after creation that a recovery point is
 deleted. Must be greater than 90 days plus MoveToColdStorageAfterNumberOfDays.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(0|[1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|[12][0-9]{4}|3[0-4][0-9]{3}|35[0-5]
[0-9]{2}|35600)$",
            "default": "0"
          },
          "maxItems": 1
        },
        "MoveToColdStorageAfterNumberOfDays": {
          "description": "The number of days after creation that a recovery point is
 moved to cold storage.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(0|[1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|[12][0-9]{4}|3[0-4][0-9]{3}|35[0-5]
[0-9]{2}|35600)$",
            "default": "0"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "BackupPlanName",
          "DeleteAfterNumberOfDays",
          "DestinationRegion",
          "DestinationVaultName",
          "DestinationEncryptionKeyArn",
          "MoveToColdStorageAfterNumberOfDays",
          "RuleName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BackupPlanName",
        "DestinationRegion",
        "RuleName"
      ]
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0g690ekkyfm79

Classifications:

- Deployment | Advanced stack components | Elastic File System (EFS) | Create from backup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EFS From Backup",
  "description": "Create an AWS Elastic File System (EFS) stack from backup.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartRestoreJobEFS.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartRestoreJobEFS"
      ],
      "default": "AWSManagedServices-StartRestoreJobEFS"
    },
    "Region": {
      "description": "The AWS Region in which the EFS snapshot is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
```

```
    "properties": {
      "BackupVaultName": {
        "description": "The name of a logical container where backups are stored.
The backup vault name is case sensitive and must contain from 2 to 50 alphanumeric
characters or hyphens.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^[a-zA-Z0-9_\\/\\-]{2,50}$"
        },
        "maxItems": 1
      },
      "EnableEncryption": {
        "description": "Flag to control, when restoring to a new filesystem, whether
it is encrypted or not. If specified, the KmsKeyId must also be set. If not specified,
the new filesystem will be created without encryption.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "maxItems": 1
      },
      "ItemsToRestore": {
        "description": "The list containing up to five directories or files paths
to be restored. Paths are case sensitive and cannot contain the following special
characters: :, *, ?, \", <, > and `. If not specified, the entire filesystem will be
restored.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^(/[^:*?\"<>`]*)$"
        },
        "maxItems": 5
      },
      "KmsKeyId": {
        "description": "The Amazon Resource Name (ARN) for the AWS KMS key to be used
to encrypt the new filesystem at rest.",
        "type": "array",
        "items": {
```

```
            "type": "string",
            "pattern": "^(|arn:aws:kms:([a-z]{2}((-gov))?-[a-z]+-\\d{1}):[0-9]{0,12}:
[a-zA-Z0-9\\_\\-\\/\\:]+)$"
          },
          "maxItems": 1
        },
        "PerformanceMode": {
          "description": "The performance mode, if restoring to a new filesystem. Use
 generalPurpose for most file systems. Use maxIO for applications where tens, hundreds,
 or thousands of EC2 instances are accessing the file system. If not specified,
 generalPurpose is used.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "generalPurpose",
              "maxIO"
            ],
            "default": "generalPurpose"
          },
          "maxItems": 1
        },
        "RecoveryPointArn": {
          "description": "The Amazon Resource Name (ARN) that uniquely identifies the
 recovery point to restore.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        },
        "RestoreToNewFileSystem": {
          "description": "Flag to control whether the restore process creates a new
 filesystem or restores it to a directory in the source filesystem. If not specified,
 it is restored to a new filesystem.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ],
```

```
            "default": "true"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "BackupVaultName",
          "EnableEncryption",
          "ItemsToRestore",
          "KmsKeyId",
          "PerformanceMode",
          "RecoveryPointArn",
          "RestoreToNewFileSystem"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BackupVaultName",
        "RecoveryPointArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0h3p576mj4rqm

Classifications:

- Management | Advanced stack components | EC2 instance stack | Change hostname (Windows)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change Windows Hostname",
  "description": "Change the hostname of an EC2 Windows instance. Note that the
 instance will be rebooted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ChangeHostname.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ChangeHostname"
      ],
      "default": "AWSManagedServices-ChangeHostname"
    },
    "Region": {
      "description": "The AWS Region where the EC2 instance is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Hostname": {
          "description": "The new hostname of the instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9-]{1,63}$"
          },
          "minItems": 1,
          "maxItems": 1
```

```
        },
        "Platform": {
          "description": "Must be windows. To change the hostname for a Linux instance,
 use CT ct-2781aqd6f6svs.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "windows",
            "enum": [
              "windows"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "Hostname",
          "Platform"
        ]
      },
      "required": [
        "InstanceId",
        "Hostname",
        "Platform"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
```

```
}
```

Schema for Change Type ct-0hahohe17csnc

Classifications:

- Management | Advanced stack components | EC2 instance stack | Encrypt instance volumes

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Encrypt Instance Volumes",
  "description": "Encrypt Elastic Block Store (EBS) volumes attached to an EC2
 instance",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-EncryptInstanceVolumes",
      "type": "string",
      "enum": [
        "AWSManagedServices-EncryptInstanceVolumes"
      ],
      "default": "AWSManagedServices-EncryptInstanceVolumes"
    },
    "Region": {
      "description": "The AWS Region where the EC2 instance is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance to encrypt volumes for. The
 instance must support encryption of EBS volumes and not part of an Auto Scaling
 group.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8}|i-[a-z0-9]{17}$"
          },
          "minItems": 1,
```

```
          "maxItems": 1
        },
        "VolumeIds": {
          "description": "The list of EBS volume IDs to encrypt. The volume IDs must be
 attached to the specified EC2 instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^vol-([0-9a-f]{8}|[0-9a-f]{17})$"
          },
          "minItems": 1,
          "maxItems": 25,
          "uniqueItems": true
        },
        "KMSKeyId": {
          "description": "The KMS key ID, or ARN, to encrypt all the new volumes.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z]{2}-[a-z]+-\\d{1}:[0-9]
{12}:key/)?([a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12}|mrk-[a-z0-9]
{32})$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DeleteStaleNonEncryptedSnapshotBackups": {
          "description": "True to delete existing snapshot backups of specified EBS
 volumes. False to not delete the existing snapshots.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "True",
              "False"
            ],
            "default": "True"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
```

```
          "InstanceId",
          "VolumeIds",
          "KMSKeyId",
          "DeleteStaleNonEncryptedSnapshotBackups"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId",
        "VolumeIds",
        "KMSKeyId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0hi7z7tyikjf6

Classifications:

- Management | Monitoring and notification | SQS | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update SQS",
  "description": "Use to modify the properties of an existing Amazon Simple Queue
 Service instance.",
  "type": "object",
  "properties": {
```

```
   "VpcId": {
     "description": "ID of the VPC that contains the SQS queue, in the form
vpc-0123abcd or vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "StackId": {
     "description": "ID of the stack instance that contains the SQS queue, in the form
stack-a1b2c3d4e5f67890e.",
     "type": "string",
     "pattern": "^stack-[a-z0-9]{17}$"
   },
   "Parameters": {
     "description": "Specifications for the stack.",
     "type": "object",
     "properties": {
       "SQSDelaySeconds": {
         "description": "The time in seconds that the delivery of new messages in the
queue will be delayed.",
         "type": "number",
         "minimum": 0,
         "maximum": 900,
         "default": 0
       },
       "SQSMaximumMessageSize": {
         "description": "The limit of how many bytes a message can contain before SQS
rejects it.",
         "type": "number",
         "minimum": 1024,
         "maximum": 262144,
         "default": 262144
       },
       "SQSMessageRetentionPeriod": {
         "description": "The number of seconds SQS retains a message, from 60 (1
minute) to 1209600 (14 days).",
         "type": "number",
         "minimum": 60,
         "maximum": 1209600,
         "default": 345600
       },
       "SQSQueueName": {
         "description": "A name for the queue.",
         "type": "string",
         "pattern": "^[a-zA-Z0-9-_]{1,80}$",
```

```
          "minLength": 1,
          "maxLength": 80
        },
        "SQSReceiveMessageWaitTimeSeconds": {
          "description": "The number of seconds that the ReceiveMessage call waits for
 a message to arrive in the queue before returning a response.",
          "type": "number",
          "minimum": 0,
          "maximum": 20,
          "default": 0
        },
        "SQSVisibilityTimeout": {
          "description": "The number of seconds that the received messages are
 hidden from subsequent retrieve requests after being retrieved by a ReceiveMessage
 request.",
          "type": "number",
          "minimum": 0,
          "maximum": 43200
        }
      }
    }
  },
  "additionalProperties": false,
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0hu3q3957aghj

Classifications:

- Deployment | Advanced stack components | ACM | Create private certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Private ACM Certificate",
  "description": "Create a private AWS Certificate Manager (ACM) certificate with email
 or DNS validation. To create a public ACM certificate, use ct-3ll9hnadql9s1.",
  "type": "object",
```

```
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RequestACMCertificateV2",
      "type": "string",
      "enum": [
        "AWSManagedServices-RequestACMCertificateV2"
      ],
      "default": "AWSManagedServices-RequestACMCertificateV2"
    },
    "Region": {
      "description": "The AWS Region in which you want the ACM certificate, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DomainName": {
          "description": "The fully qualified domain name (FQDN), such as
 www.example.com, that you want to secure with an ACM certificate.",
          "type": "string",
          "pattern": "^(?!://)(?=.{1,255}$)((.{1,63}\\.){1,127}(?![0-9]*$)[a-z0-9-]+\
\.?)$"
        },
        "CertificateType": {
          "description": "Confirm that you are creating a private ACM certificate. To
 create a public ACM certificate, use ct-3ll9hnadql9s1.",
          "type": "string",
          "enum": [
            "Private"
          ],
          "default": "Private"
        },
        "CertificateAuthorityArn": {
          "description": "The Amazon Resource Name (ARN) of the private certificate
 authority (CA) used to issue the certificate.",
          "type": "string",
          "pattern": "^arn:aws:.+$"
        },
        "SubjectAlternativeNames": {
          "description": "Additional FQDNs to be included in the subject alternative
 name extension of the ACM certificate.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^(?!://)(?=.{1,255}$)((.{1,63}\\.){1,127}(?![0-9]*$)[a-z0-9-]+\
\.?)$"
          },
          "minItems": 1,
          "maxItems": 5
        },
        "Route53DNSValidation": {
          "description": "True for automatic ACM validation using your Route53 DNS, if
 the ACM and the domain are on the same account; false for no automatic validation.
 Default is false.",
          "type": "string",
          "enum": [
            "True",
            "False"
          ],
          "default": "False"
        }
      },
      "metadata": {
        "ui:order": [
          "DomainName",
          "CertificateType",
          "CertificateAuthorityArn",
          "SubjectAlternativeNames",
          "Route53DNSValidation"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DomainName",
        "CertificateAuthorityArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
```

```
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0idxb0xsg1ui6

Classifications:

- Management | Advanced stack components | RDS snapshot | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete RDS Snapshots",
  "description": "Delete DB instance or cluster snapshots. This document only supports
 deletion of 'manual' and 'awsbackup' snapshot types. If the snapshot is being copied,
 the copy operation is terminated. The snapshot must be in available state to be
 deleted. If one or more snapshots cannot be deleted, automation fails. Up to 20
 snapshots can be deleted in one execution.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteRDSSnapshotsV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteRDSSnapshotsV2"
      ],
      "default": "AWSManagedServices-DeleteRDSSnapshotsV2"
    },
    "Region": {
      "description": "The AWS Region where the DB snapshots are located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SnapshotNamesOrArns": {
          "description": "A list of up to 20 RDS snapshot names or ARN's to delete.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!rds:).*$"
          },
          "minItems": 1,
          "maxItems": 20
        }
      },
      "metadata": {
        "ui:order": [
          "SnapshotNamesOrArns"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SnapshotNamesOrArns"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0ikpop8zqhkxg

Classifications:

- Management | Access | Stack admin access | Update

```
{
```

```
 "$schema": "http://json-schema.org/draft-04/schema#",
 "name": "Grant stack admin access",
 "description": "Request admin access for one or more users for one or more stacks.
The maximum access time is 12 hours.",
 "type": "object",
 "properties": {
   "DomainFQDN": {
     "description": "The FQDN for the user accounts to grant access to.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "StackIds": {
     "description": "A minimum of one stack ID is required.",
     "type": "array",
     "items": {
       "type": "string",
       "pattern": "^stack-[a-z0-9]{17}$|^SC-[0-9]{12}-pp-[a-zA-Z0-9]{13}$"
     },
     "minItems": 1,
     "uniqueItems": true
   },
   "TimeRequestedInHours": {
     "description": "The amount of time, in hours, requested for access to the
instance. Access is terminated after this time.",
     "type": "integer",
     "minimum": 1,
     "default": 1
   },
   "Usernames": {
     "description": "One or more Active Directory user names used to grant access.",
     "type": "array",
     "items": {
       "type": "string"
     },
     "minItems": 1,
     "uniqueItems": true
   },
   "VpcId": {
     "description": "The ID of the VPC that contains the stacks where access is
required, in the form of vpc-12345678 or vpc-1234567890abcdef0.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   }
```

```
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackIds",
      "Usernames",
      "DomainFQDN",
      "TimeRequestedInHours"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DomainFQDN",
    "StackIds",
    "Usernames",
    "VpcId"
  ]
}
```

Schema for Change Type ct-0ixp4ch2tiu04

Classifications:

- Management | Applications | IAM instance profile | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create IAM instance profile",
  "description": "Use to create an instance profile.",
  "type": "object",
  "properties": {
    "InstanceProfileDescription": {
      "description": "The description of the instance profile.",
      "type": "string",
      "maxLength": 5000
    },
    "InstanceProfileName": {
      "description": "The name of the instance profile to create.",
      "type": "string",
      "minLength": 1,
      "maxLength": 128,
      "pattern": "^[a-zA-Z0-9_.=@,+-]{1,128}$"
```

```
    },
    "RelatedIds": {
      "description": "(Optional) IDs of resources related to the change request.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 1000,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "InstanceProfileDescription",
    "InstanceProfileName"
  ],
  "metadata": {
    "ui:order": [
      "InstanceProfileDescription",
      "InstanceProfileName",
      "RelatedIds",
      "Priority"
    ]
  }
}
```

Schema for Change Type ct-0jb01cofkhwk1

Classifications:

- Management | Managed account | Stack access duration | Override (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Override Stack Access Duration",
  "description": "Use to override maximum stack access time for all stacks in this
 account for single landing zone (SALZ) and for all stacks of the member accounts of
 an organization for multi-landing zone (MALZ). For multi-landing zone (MALZ), please
 raise a request for change (RFC) from shared-services account with this change type
 (CT) ID. Access can be overridden from a minimum of 1 hour to a maximum of 120 hours,
 default stack access is granted for 12 hours.",
  "type": "object",
  "properties": {
    "TimeRequestedInHours": {
      "description": "The amount of time, in hours, requested to override. Access can
 be overridden from a minimum of 1 hour to a maximum of 120 hours, default stack access
 is granted for 12 hours. Access is terminated after this time.",
      "type": "integer",
      "minimum": 1,
      "maximum": 120,
      "default": 1
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "TimeRequestedInHours",
      "Priority"
    ]
  },
  "required": [
    "TimeRequestedInHours"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0k4b96aatyqgl

Classifications:

- Management | Advanced stack components | Tag | Bulk update (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Bulk Update Resource Tags (Review Required)",
  "description": "Bulk add tags to existing, supported resources except those
 in AMS infrastructure stacks (stacks named mc-*). Tags simplify categorization,
 identification and targeting AWS resources. Use this with AWS Tag Editor when managing
 large numbers of tags (i.e. >50). For Autoscaling, EC2, Elastic Load Balancing, RDS
 resources and S3 buckets, use automated CT ct-3047c34zuvswh.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the tag operation.",
      "type": "string",
      "maxLength": 5000
    },
    "CsvS3Url": {
      "description": "The S3 bucket endpoint for the CSV file with the tag update
 details. The CSV file must be formatted to the correct format. Please see AMS tag
 documentation for the correct format of the CSV file.",
      "type": "string",
      "pattern": "^https?://[a-z0-9]([-.a-z0-9]+)[a-z0-9]\\.s3\\.((([a-z]{2}-[a-z]+-\
\d{1}\\.)?))amazonaws\\.com/[\\S]*",
      "minLength": 1,
      "maxLength": 1536
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
```

```
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Description",
      "CsvS3Url",
      "Priority"
    ]
  },
  "required": [
    "Description",
    "CsvS3Url"
  ]
}
```

Schema for Change Type ct-0kbey7hb00atp

Classifications:

- Deployment | Patching | SSM patch baseline | Create (Windows)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Baseline (Windows)",
  "description": "Create an AWS Systems Manager (SSM) patch baseline to define which
 patches are approved for installation on your instances for Windows OS. Specify
 existing instance \"Patch Group\" tag values for the patch baseline. The patch
 baseline is an SSM resource that you can manage with the SSM console.",
  "additionalProperties": false,
  "properties": {
    "ApprovalRules": {
      "description": "Create auto-approval rules to specify that certain types of
 operating system patches are approved automatically.",
      "items": {
        "additionalProperties": false,
        "properties": {
          "ApproveAfterDays": {
            "default": 7,
            "description": "The number of days to wait after a patch is released before
 approving patches automatically.",
            "maximum": 100,
            "minimum": 0,
            "type": "integer"
```

```
          },
          "Classification": {
            "description": "The Classification of the patches to be selected. Allowed
 values are \"CriticalUpdates\", \"DefinitionUpdates\", \"Drivers\", \"FeaturePacks
\", \"SecurityUpdates\", \"ServicePacks\", \"Tools\", \"UpdateRollups\", \"Updates\",
 \"Upgrades\" and \"All\".",
            "items": {
              "enum": [
                "CriticalUpdates",
                "DefinitionUpdates",
                "Drivers",
                "FeaturePacks",
                "SecurityUpdates",
                "ServicePacks",
                "Tools",
                "UpdateRollups",
                "Updates",
                "Upgrades",
                "All"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          },
          "Severity": {
            "description": "The severity of the patches to be selected. Allowed values
 are \"Critical\", \"Important\", \"Low\", \"Moderate\", \"Unspecified\" and \"All
\".",
            "items": {
              "enum": [
                "Critical",
                "Important",
                "Low",
                "Moderate",
                "Unspecified",
                "All"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          }
        },
```

```
        "metadata": {
          "ui:order": [
            "Severity",
            "Classification",
            "ApproveAfterDays"
          ]
        },
        "required": [
          "ApproveAfterDays"
        ],
        "type": "object"
      },
      "maxItems": 10,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "ApprovedPatches": {
      "description": "The list of patches to approve explicitly.",
      "items": {
        "type": "string",
        "maxLength": 100,
        "minLength": 1,
        "pattern": "^(^KB[0-9]{1,7}$)|(^MS[0-9]{2}-[0-9]{3}$)"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Description": {
      "description": "A meaningful description for this patch baseline.",
      "maxLength": 500,
      "minLength": 1,
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch baseline.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "OperatingSystem": {
```

```
      "default": "Windows",
      "description": "The operating system of instances to which this baseline is
applied.",
      "enum": [
        "Windows"
      ],
      "type": "string"
    },
    "PatchGroupTagValues": {
      "description": "A list of the values of your \"Patch Group\" tags on the
instances you want patched; the values for up to twenty-five \"Patch Group\" tags can
be provided. Instances with those values are associated with this patch baseline.",
      "items": {
        "maxLength": 256,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 25,
      "minItems": 1,
      "type": "array",
      "uniqueItems": true
    },
    "RejectedPatches": {
      "description": "The list of patches to reject explicitly.",
      "items": {
        "maxLength": 100,
        "minLength": 1,
        "pattern": "^(^KB[0-9]{1,7}$)|(^MS[0-9]{2}-[0-9]{3}$)",
        "type": "string"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the SSM patch
baseline resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
```

```
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "OperatingSystem",
      "Name",
      "Description",
      "PatchGroupTagValues",
      "ApprovalRules",
      "ApprovedPatches",
      "RejectedPatches",
      "Tags"
    ]
  },
  "required": [
    "Name",
    "PatchGroupTagValues",
    "OperatingSystem"
  ],
  "type": "object"
```

```
}
```

Schema for Change Type ct-0loed9dzig1ze

Classifications:

- Management | Advanced stack components | RDS database stack | Update Storage

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update RDS Storage",
  "description": "Change the RDS instance storage type, capacity or IOPS through direct
 API calls. The RDS instance can be standalone or belong to a CloudFormation stack, in
 the latter case, the change might cause stack drift. To avoid causing stack drift,
 please use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS instance was
 provisioned via CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateRDSStorage.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateRDSStorage"
      ],
      "default": "AWSManagedServices-UpdateRDSStorage"
    },
    "Region": {
      "description": "The AWS Region of the account with the RDS database instance; for
 example, us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBInstanceIdentifier": {
          "description": "The identifier of the RDS database instance; for example,
 mydbinstance.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^(?!(mc|ams|awsms)-)[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]
{0,62}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "AllocatedStorage": {
          "description": "The new amount of storage in gibibytes (GiB) to allocate for
 the DB instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^\\d+$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "StorageType": {
          "description": "The storage type to be associated with the DB instance.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "",
              "gp2",
              "gp3",
              "io1",
              "Magnetic"
            ],
            "default": ""
          }
        },
        "Iops": {
          "description": "The new provisioned IOPS (I/O operations per second) value
 for the RDS instance. This parameter is only valid for io1 and gp3 storage type.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^\\d+$",
            "default": ""
          }
        },
        "ApplyImmediately": {
```

```
          "description": "True to apply the change immediately, false to schedule the
 change on next maintenance window. To discover your next maintenance window, check the
 details page for the instance in the RDS console.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "DBInstanceIdentifier",
          "AllocatedStorage",
          "StorageType",
          "Iops",
          "ApplyImmediately"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBInstanceIdentifier",
        "ApplyImmediately"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0lqruajvhwsbk

Classifications:

- Management | Advanced stack components | Security group | Authorize egress rule


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Authorize Egress Rule",
  "description": "Authorize the egress rule for the specified security group (SG). You
 must specify the configurations of the egress rule that you are authorizing. Note that
 this adds an egress rule to the specified SG but does not modify any existing egress
 rules.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AuthorizeSecurityGroupEgressRule",
      "type": "string",
      "enum": [
        "AWSManagedServices-AuthorizeSecurityGroupEgressRule"
      ],
      "default": "AWSManagedServices-AuthorizeSecurityGroupEgressRule"
    },
    "Region": {
      "description": "The AWS Region in which the security group is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityGroupId": {
          "description": "The ID of the security group (SG) that you are updating, in
 the form sg-0123456789abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
```

```
        },
        "IpProtocol": {
          "description": "The IP protocol name, or IP protocol number, for the egress
 rule. For example, for TCP, enter either TCP, or (IP protocol number) 6. If you enter
 ICMP, you can specify any or all of the ICMP types and codes.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\+-\\\\(\\\\)\\w]{1,18}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "FromPort": {
          "description": "Start of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ToPort": {
          "description": "End of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Destination": {
          "description": "An IP address, in the form 255.255.255.255, or an IP address
 range in CIDR notation, in the form 255.255.255.255/32, or the ID of another security
 group in the same region; or self to specify the same security group.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8}$|^sg-[0-9a-f]{17}$|^self$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Description": {
          "description": "A meaningful description of the egress rule.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^[ a-zA-Z0-9._\\-:/()#,@\\[\\]+=&;{}!$\\*]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SecurityGroupId",
          "IpProtocol",
          "FromPort",
          "ToPort",
          "Destination",
          "Description"
        ]
      },
      "required": [
        "SecurityGroupId",
        "IpProtocol",
        "FromPort",
        "ToPort",
        "Destination"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
```

```
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0ltm873rsebx9

Classifications:

- Management | Advanced stack components | Load balancer (ELB) stack | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update load balancer (ELB) stack",
  "description": "Modify the properties of an existing Amazon ELB Classic Load Balancer
 created using CT id ct-12amsdz909cfh, version 3.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the ELB that you are updating, in the form stack-
a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the ELB.",
      "type": "object",
      "properties": {
        "ELBSubnetIds": {
          "description": "One or more subnet IDs for the load balancer, in the form
 subnet-0123abcd or subnet-01234567890abcdef. Changing this value during an update
```

```
does not append to the existing subnets associated with the load balancer. Include all
required subnets when modifying this value.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
         },
         "minItems": 1,
         "uniqueItems": true
       },
       "ELBBackendInstances": {
         "description": "One or more EC2 instance IDs to associate with the load
balancer, in the form of i-0123abcd or i-01234567890abcdef for a single instance,
or i-0123abcd,i-12345abcd or i-01234567890abcdef,i-2345678901abcdefg for multiple
instances. A load balancer can be associated with an autoscaling group by specifying
the load balancer name in the ASGLoadBalancerNames property during creation or update
of the autoscaling group. Changing this value during an update does not append to
the existing instances associated with the load balancer. Include all required EC2
instances not part of an autoscaling group when modifying this value. To remove all
EC2 instances not part of an autoscaling group during an update specify None.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^i-[a-z0-9]{8}$|^i-[a-z0-9]{17}$|^[Nn]one$|^$"
         },
         "minItems": 1,
         "uniqueItems": true
       },
       "ELBCrossZone": {
         "description": "With cross-zone load balancing, your load balancer nodes
route traffic to the back-end instances across all Availability Zones. True to enable,
false to disable. The default is true.",
         "type": "boolean"
       },
       "ELBCookieExpirationPeriod": {
         "description": "The time period, in seconds, after which the cookie is
considered stale. If this parameter isn't specified, the sticky session lasts for the
duration of the browser session.",
         "type": "string",
         "pattern": "^[0-9]+$|^$"
       },
       "ELBCookieExpirationPeriod2": {
```

```
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "type": "string",
          "pattern": "^[0-9]+$|^$"
        },
        "ELBCookieStickinessPolicyName": {
          "description": "A name for the cookie stickiness policy. The name must be
 unique within the set of policies for this load balancer.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "ELBCookieStickinessPolicyName2": {
          "description": "A name for the second cookie stickiness policy. The name must
 be unique within the set of policies for this load balancer.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "ELBHealthCheckHealthyThreshold": {
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "type": "number",
          "minimum": 2,
          "maximum": 10
        },
        "ELBHealthCheckInterval": {
          "description": "The approximate interval, in seconds, between health
 checks.",
          "type": "number",
          "minimum": 5,
          "maximum": 300
        },
        "ELBHealthCheckTarget": {
          "description": "The protocol, port, and path of the instance to check. For
 example, HTTP:80/weather/us/wa/seattle. The protocol can be TCP, HTTP, HTTPS, or SSL.
 The range of valid ports is 1 through 65535.",
          "type": "string",
          "pattern": "^(HTTP|HTTPS):[0-9]{1,5}[/][a-zA-Z0-9/_.-]*$|^(SSL|TCP):[0-9]
{1,5}$"
        },
        "ELBHealthCheckTimeout": {
          "description": "The amount of time, in seconds, to wait for a response to a
 health check. Must be less than the value for ELBHealthCheckInterval.",
          "type": "number",
```

```
          "minimum": 2,
          "maximum": 60
        },
        "ELBHealthCheckUnhealthyThreshold": {
          "description": "The number of consecutive health check failures required to
 declare an EC2 instance unhealthy.",
          "type": "number",
          "minimum": 2,
          "maximum": 10
        },
        "ELBIdleTimeout": {
          "description": "The time, in seconds, that a connection to the load balancer
 can remain idle, which means no data is sent over the connection. After the specified
 time, the load balancer closes the connection.",
          "type": "number",
          "minimum": 1,
          "maximum": 3600
        },
        "ELBInstancePort": {
          "description": "The TCP port the listener uses to send traffic to the target
 instance. Changing this value during an update will cause the existing listener to be
 deleted and a new one created. Clients will be unable to connect during this time.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBInstancePort2": {
          "description": "The TCP port the optional second listener uses to send
 traffic to the target instance. Changing this value during an update will cause the
 existing listener to be deleted and a new one created. Clients will be unable to
 connect during this time.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBInstanceProtocol": {
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance). Changing this value during an
 update will cause the existing listener to be deleted and a new one created. Clients
 will be unable to connect during this time.",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
```

```
          ]
        },
        "ELBInstanceProtocol2": {
          "description": "The protocol the second listener uses for routing traffic
to back-end connections (load balancer to backend instance). Changing this value
during an update will cause the existing listener to be deleted and a new one created.
Clients will be unable to connect during this time.",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "ELBLoadBalancerPort": {
          "description": "The port number for the load balancer to use when routing
external incoming traffic. Changing this value during an update will cause the
existing listener to be deleted and a new one created. Clients will be unable to
connect during this time.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBLoadBalancerPort2": {
          "description": "The port number for the load balancer to use when routing
external incoming traffic on the second listener. Changing this value during an update
will cause the existing listener to be deleted and a new one created. Clients will be
unable to connect during this time.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBLoadBalancerProtocol": {
          "description": "The transport protocol to use for routing front-end
connections (client to load balancer). Changing this value during an update will cause
the existing listener to be deleted and a new one created. Clients will be unable to
connect during this time.",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
```

```
        "ELBLoadBalancerProtocol2": {
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) on the second listener. Changing this value
 during an update will cause the existing listener to be deleted and a new one created.
 Clients will be unable to connect during this time.",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "ELBSSLCertificateId": {
          "description": "The Amazon Resource Name (ARN)
 of the SSL certificate to use, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012. This must be
 specified if the HTTPS or SSL protocol is specified for ELBLoadBalancerProtocol.
 Changing this value during an update will cause the existing listener to be deleted
 and a new one created. Clients will be unable to connect during this time.",
          "type": "string",
          "pattern": "^$|(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})|(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$|^([a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]
{12})$"
        },
        "ELBSSLCertificateId2": {
          "description": "The Amazon Resource Name (ARN) of the SSL certificate
 to use for the optional second listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012. Required only
 if a second listener is used and ELBLoadBalancerProtocol2 is either HTTPS or SSL.
 Changing this value during an update will cause the existing listener to be deleted
 and a new one created. Clients will be unable to connect during this time.",
          "type": "string",
          "pattern": "^$|(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})|(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$|^([a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]
{12})$"
        }
      },
      "metadata": {
        "ui:order": [
          "ELBSubnetIds",
          "ELBBackendInstances",
```

```
          "ELBIdleTimeout",
          "ELBCrossZone",
          "ELBHealthCheckTarget",
          "ELBHealthCheckInterval",
          "ELBHealthCheckTimeout",
          "ELBHealthCheckHealthyThreshold",
          "ELBHealthCheckUnhealthyThreshold",
          "ELBCookieStickinessPolicyName",
          "ELBCookieExpirationPeriod",
          "ELBInstancePort",
          "ELBInstanceProtocol",
          "ELBLoadBalancerPort",
          "ELBLoadBalancerProtocol",
          "ELBSSLCertificateId",
          "ELBCookieExpirationPeriod2",
          "ELBCookieStickinessPolicyName2",
          "ELBInstancePort2",
          "ELBInstanceProtocol2",
          "ELBLoadBalancerPort2",
          "ELBLoadBalancerProtocol2",
          "ELBSSLCertificateId2"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0mss4i7neuj7f

Classifications:

- Management | Managed Firewall | Outbound (Palo Alto) | Update security policy


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Security Policy",
  "description": "Update a security policy for AMS managed Palo Alto firewall -
 Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be UpdateSecurityPolicy.",
      "type": "string",
      "enum": [
        "UpdateSecurityPolicy"
      ],
      "default": "UpdateSecurityPolicy"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityPolicyName": {
          "description": "The name of the security policy. Must start with custom-
sec-.",
          "type": "string",
          "pattern": "^custom-sec-[a-zA-Z0-9][a-zA-Z0-9-_]{0,51}$"
        },
        "SourceAddressesToAdd": {
          "description": "A list of source addresses to add to the policy.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "DestinationAddressesToAdd": {
```

```
          "description": "A list of destination addresses to add to the policy. Supply
 values for this parameter or for AllowListsToAdd, but not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)|((([a-zA-
Z0-9][a-zA-Z0-9-_]{0,62}[a-zA-Z0-9]{0,1}))\\.){1,127}([a-zA-Z][a-zA-Z0-9\\-]{0,23}[a-
zA-Z]))$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "AllowListsToAdd": {
          "description": "A list of allowlists to add to the policy. Supply values for
 this parameter or for DestinationAddressesToAdd, but not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
          },
          "minItems": 1,
          "maxItems": 10
        },
        "ServicePortsToAdd": {
          "type": "object",
          "description": "A list of Transmission Control Protocol (TCP) and User
 Datagram Protocol (UDP) service ports to add.",
          "properties": {
            "TCPPortsToAdd": {
              "description": "A list of Transmission Control Protocol (TCP) service
 ports to add.",
              "type": "array",
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            },
            "UDPPortsToAdd": {
              "description": "A list of User Datagram Protocol (UDP) service ports to
 add.",
              "type": "array",
```

```
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            }
          },
          "metadata": {
            "ui:order": [
              "TCPPortsToAdd",
              "UDPPortsToAdd"
            ]
          }
        },
        "SourceAddressesToRemove": {
          "description": "A list of source addresses to remove from the policy.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "DestinationAddressesToRemove": {
          "description": "A list of destination addresses to remove from the policy.
 Supply values for this parameter or for AllowListsToRemove, but not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)|((([a-zA-
Z0-9][a-zA-Z0-9-_]{0,62}[a-zA-Z0-9]{0,1}))\\.){1,127}([a-zA-Z][a-zA-Z0-9\\-]{0,23}[a-
zA-Z]))$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "AllowListsToRemove": {
          "description": "A list of allowlists to remove from the policy. Supply values
 for this parameter or for DestinationAddressesToRemove, but not both.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
          },
          "minItems": 1,
          "maxItems": 10
        },
        "ServicePortsToRemove": {
          "type": "object",
          "description": "A list of Transmission Control Protocol (TCP) and User
Datagram Protocol (UDP) service ports to remove.",
          "properties": {
            "TCPPortsToRemove": {
              "description": "A list of Transmission Control Protocol (TCP) service
ports to remove.",
              "type": "array",
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            },
            "UDPPortsToRemove": {
              "description": "A list of User Datagram Protocol (UDP) service ports to
remove.",
              "type": "array",
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            }
          },
          "metadata": {
            "ui:order": [
              "TCPPortsToRemove",
              "UDPPortsToRemove"
            ]
          }
        },
        "ActionType": {
```

```
          "description": "The type of action the security policy will perform on
 outbound traffic that matches the policy's rules.",
          "type": "string",
          "enum": [
            "Allow",
            "Deny"
          ]
        },
        "EnablePolicy": {
          "description": "True to enable the security policy, false to disable it.",
          "type": "boolean"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "SecurityPolicyName",
          "SourceAddressesToAdd",
          "DestinationAddressesToAdd",
          "AllowListsToAdd",
          "ServicePortsToAdd",
          "SourceAddressesToRemove",
          "DestinationAddressesToRemove",
          "AllowListsToRemove",
          "ServicePortsToRemove",
          "ActionType",
          "EnablePolicy"
        ]
      },
      "required": [
        "SecurityPolicyName"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "RequestType",
      "Parameters"
    ]
  },
  "required": [
    "RequestType",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-0o4zi9bzg74lp

Classifications:

- Management | Advanced stack components | S3 storage | Add event notification

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Event Notification",
  "description": "Add an event notification to the specified S3 bucket through direct
 API calls. The S3 bucket can be standalone or belong to a CloudFormation stack. For
 buckets in CloudFormation stacks, be aware that stack drift might occur if the bucket
 was provisioned through CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddBucketEventNotification.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddBucketEventNotification"
      ],
      "default": "AWSManagedServices-AddBucketEventNotification"
    },
    "Region": {
      "description": "The AWS Region in which the source bucket is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BucketName": {
          "description": "The name of the bucket for which to add the notification
 configuration.",
          "type": "string",
          "pattern": "^(?!(mc|ams|awsms)-)[a-z0-9][-.a-z0-9]{1,61}[a-z0-9]$"
        },
        "EventName": {
```

```
          "description": "A unique identifier for the event notification
 configuration.",
          "type": "string",
          "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,255}$"
        },
        "Prefix": {
          "description": "The object key name prefix to which the filtering rule
 applies. If a value is specified, event notifications will be limited to objects with
 key starting with the specified characters.",
          "type": "string",
          "pattern": "^.{0,1024}$",
          "default": ""
        },
        "Suffix": {
          "description": "The object key name suffix to which the filtering rule
 applies. If a value is specified, event notifications will be limited to objects with
 key ending with the specified characters.",
          "type": "string",
          "pattern": "^.{0,1024}$",
          "default": ""
        },
        "EventTypes": {
          "description": "Specify the events for which you want to receive
 notifications. Enter '*' if you would like to enable notifications for all available
 event types or if selecting EventBridge as the destination. Refer to https://
docs.aws.amazon.com/AmazonS3/latest/userguide/notification-how-to-event-types-and-
destinations.html#supported-notification-event-types for details on the values.",
          "type": "array",
          "items": {
            "enum": [
              "s3:ObjectCreated:*",
              "s3:ObjectCreated:Put",
              "s3:ObjectCreated:Post",
              "s3:ObjectCreated:Copy",
              "s3:ObjectCreated:CompleteMultipartUpload",
              "s3:ObjectRemoved:*",
              "s3:ObjectRemoved:Delete",
              "s3:ObjectRemoved:DeleteMarkerCreated",
              "s3:ObjectRestore:*",
              "s3:ObjectRestore:Post",
              "s3:ObjectRestore:Completed",
              "s3:ObjectRestore:Delete",
              "s3:ReducedRedundancyLostObject",
              "s3:Replication:*",
```

```
              "s3:Replication:OperationFailedReplication",
              "s3:Replication:OperationMissedThreshold",
              "s3:Replication:OperationReplicatedAfterThreshold",
              "s3:Replication:OperationNotTracked",
              "s3:LifecycleExpiration:*",
              "s3:LifecycleExpiration:Delete",
              "s3:LifecycleExpiration:DeleteMarkerCreated",
              "s3:LifecycleTransition",
              "s3:IntelligentTiering",
              "s3:ObjectTagging:*",
              "s3:ObjectTagging:Put",
              "s3:ObjectTagging:Delete",
              "s3:ObjectAcl:Put",
              "*"
            ],
            "type": "string"
          },
          "minItems": 1,
          "maxItems": 27
        },
        "DestinationARN": {
          "description": "The Amazon Resource Name (ARN) of the Amazon SQS queue, the
 Amazon SNS topic or the Lambda function to which Amazon S3 publishes a message when
 it detects events of the specified type. Input 'eventbridge' for using EventBridge as
 destination.",
          "type": "string",
          "pattern": "(^arn:(aws|aws-cn|aws-us-gov):(lambda|sns|sqs):\\w{2}-[a-z]+-\
\d{1}:\\d{12}:((function:[a-zA-Z0-9-_]{1,64})|([a-zA-Z0-9-_.]{1,256}))|eventbridge)$"
        }
      },
      "metadata": {
        "ui:order": [
          "BucketName",
          "EventName",
          "EventTypes",
          "DestinationARN",
          "Prefix",
          "Suffix"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BucketName",
        "EventName",
```

```
        "EventTypes",
        "DestinationARN"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0pgvtw5rpcsb6

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create from backup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS From Backup",
  "description": "Create an Amazon Relational Database Service (RDS) from a backup.
 When you restore a backup this way, the service-specific restore parameters are
 presented automatically.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
```

```
      "description": "ID of the VPC where the backup is stored, in the form
 vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-siqajx00000000000.",
      "type": "string",
      "enum": [
        "stm-siqajx00000000000"
      ]
    },
    "Name": {
      "description": "A name for the stack; this becomes the Stack Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
```

```
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "DBInstanceClass": {
          "description": "The compute and memory capacity for the DB instance. To
inherit this value from the backup, use inherit.",
          "type": "string",
          "enum": [
            "inherit",
            "db.m1.small",
            "db.m1.medium",
            "db.m1.large",
            "db.m1.xlarge",
            "db.m2.xlarge",
            "db.m2.2xlarge",
            "db.m2.4xlarge",
            "db.m3.medium",
            "db.m3.large",
            "db.m3.xlarge",
            "db.m3.2xlarge",
            "db.m4.large",
            "db.m4.xlarge",
            "db.m4.2xlarge",
            "db.m4.4xlarge",
            "db.m4.10xlarge",
```

```
            "db.m4.16xlarge",
            "db.r3.large",
            "db.r3.xlarge",
            "db.r3.2xlarge",
            "db.r3.4xlarge",
            "db.r3.8xlarge",
            "db.r4.large",
            "db.r4.xlarge",
            "db.r4.2xlarge",
            "db.r4.4xlarge",
            "db.r4.8xlarge",
            "db.r4.16xlarge",
            "db.t1.micro",
            "db.t2.micro",
            "db.t2.small",
            "db.t2.medium",
            "db.t2.large",
            "db.t2.xlarge",
            "db.t2.2xlarge"
          ],
          "default": "inherit"
        },
        "DBInstanceIdentifier": {
          "description": "A name for the DB instance. If you specify a name, it is
converted to lowercase. If you don't specify a name, a unique physical ID is generated
and used for the DBInstanceIdentifier.",
          "type": "string",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$",
          "default": ""
        },
        "DBSnapshotIdentifier": {
          "description": "The name of the RDS DB backup to use, in the form
awsbackup:job-00000000-0000-0000-0000-000000000000.",
          "type": "string"
        },
        "DBSubnetIds": {
          "description": "Two or more subnet IDs for the DB instance, in the form
subnet-0123abcd or subnet-01234567890abcdef, spanning at least two Availability
Zones.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
```

```
          "minItems": 2,
          "maxItems": 20,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "DBInstanceClass",
          "DBInstanceIdentifier",
          "DBSnapshotIdentifier",
          "DBSubnetIds"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBSnapshotIdentifier",
        "DBSubnetIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0q0bic0ywqk6c

Classifications:

- Management | Advanced stack components | Stack | Delete
- Management | Standard stacks | Stack | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete stack",
  "description": "Delete an existing stack and its resources from your account.
 The effects of deleting a resource vary. For details, see the appropriate AWS
 documentation for the resource. Note that termination protection on a resource in the
 stack causes the RFC to fail. To check for a resource's termination protection status,
 see the corresponding AWS console.",
  "type": "object",
  "properties": {
    "StackId": {
      "description": "The ID of the stack instance to delete, in the form stack-
a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution
 of deleting the stack. This does not prolong the execution. If the delete is not
 completed in the specified time, the RFC is failed and you are notified that the
 delete is over time but continuing. The delete operation continues because delete
 operations cannot be rolled back. Set this timeout so you get notice of delete stack
 problems in a timely manner. Defaults to 60 if not provided.",
      "type": "number",
      "minimum": 0,
      "maximum": 720
    }
  },
  "additionalProperties": false,
  "required": [
    "StackId"
  ]
}
```

Schema for Change Type ct-0q43l40hxrzum

Classifications:

- Deployment | Advanced stack components | Redshift | Create (cluster subnet group)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Redshift cluster subnet group",
  "description": "Use to create a Redshift cluster subnet group.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
```

```
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-5rsvv3l4760usboci",
      "type": "string",
      "enum": [
        "stm-5rsvv3l4760usboci"
      ],
      "default": "stm-5rsvv3l4760usboci"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SubnetGroupDescription": {
          "type": "string",
```

```
          "description": "A description to help identify your cluster subnet group.",
          "minLength": 1,
          "maxLength": 255
        },
        "SubnetIds": {
          "type": "array",
          "minItems": 2,
          "uniqueItems": true,
          "description": "Two or more subnet IDs for the cluster subnet group, in the
form subnet-0123abcd or subnet-01234567890abcdef, spanning at least two Availability
Zones.",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          }
        }
      },
      "metadata": {
        "ui:order": [
          "SubnetGroupDescription",
          "SubnetIds"
        ]
      },
      "required": [
        "SubnetGroupDescription",
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
```

```
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0qbikxr9okwvy

Classifications:

- Deployment | Advanced stack components | VPN Gateway | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create VPN Gateway",
  "description": "Create a virtual private network (VPN) gateway (the endpoint on the
 VPC side of your VPN connection), and associate it to an existing virtual private
 cloud (VPC) in your account.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
```

```
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-mcti3bha1vhon1sie",
      "type": "string",
      "enum": [
        "stm-mcti3bha1vhon1sie"
      ],
      "default": "stm-mcti3bha1vhon1sie"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
  the change. This will not prolong execution, but the RFC fails if the change is not
  completed in the specified time.",
```

```
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VpcId": {
          "type": "string",
          "description": "The VPC ID to associate the VPN Gateway to.",
          "pattern": "^vpc-[0-9a-z]{17}|vpc-[0-9a-z]{8}$"
        },
        "AmazonSideAsn": {
          "type": "integer",
          "description": "The private Autonomous System Number (ASN) for the Amazon
side of a Border Gateway Protocol (BGP) session.",
          "default": 64512
        },
        "Name": {
          "type": "string",
          "description": "The tag Key name of the new VPN Gateway.",
          "pattern": "^[a-zA-Z0-9._-]+$",
          "minLength": 1,
          "maxLength": 255
        }
      },
      "metadata": {
        "ui:order": [
          "VpcId",
          "AmazonSideAsn",
          "Name"
        ]
      },
      "required": [
        "VpcId"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
```

```
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "StackTemplateId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0rmgrnr9w8mzh

Classifications:

- Management | Managed landing zone | Networking account | Remove TGW static route

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove TGW Static Route",
  "description": "Remove the specified TGW static route from the specified transit
 gateway (TGW) route table. Use this multi-account landing zone (MALZ) change type only
 in a Networking account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RemoveRouteFromTGWRouteTable.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveRouteFromTGWRouteTable"
      ],
      "default": "AWSManagedServices-RemoveRouteFromTGWRouteTable"
    },
    "Region": {
      "description": "The AWS Region of the account.",
```

```
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "TransitGatewayRouteTableId": {
          "description": "The ID of the TGW route table.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-rtb-[a-z0-9]{17}$"
          },
          "maxItems": 1
        },
        "DestinationCidrBlock": {
          "description": "The IPV4 CIDR range used for destination matches.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2])){0,1}$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "TransitGatewayRouteTableId",
          "DestinationCidrBlock"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayRouteTableId",
        "DestinationCidrBlock"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0tmpmp1wpgkr9

Classifications:

- Management | Advanced stack components | EC2 instance stack | Update instance detailed monitoring

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Detailed Monitoring",
  "description": "Update EC2 instances' detailed monitoring setting through direct API
 calls. The EC2 instances can be standalone or belong to a CloudFormation stack; in the
 latter case, the change might cause stack drift. To avoid causing stack drift, please
 use ct-38s4s4tm4ic4u instead, or ct-361tlo1k7339x if the EC2 instance was provisioned
 via CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateInstanceEnhancedMonitoring.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateInstanceEnhancedMonitoring"
      ],
      "default": "AWSManagedServices-UpdateInstanceEnhancedMonitoring"
    },
    "Region": {
      "description": "The AWS Region in which the EC2 instance is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceIds": {
          "description": "A list of up to 50 EC2 instance IDs, in the form
 i-1234567890abcdef0 or i-b188560f.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "MonitoringValue": {
          "description": "Enabled to turn on detailed monitoring for your instances.
 Disabled to turn off detailed monitoring for your instances and set it to basic
 monitoring. EC2 detailed monitoring provides more frequent metrics, published at
```

```
  one-minute intervals, instead of the five-minute intervals used in Amazon EC2 basic
  monitoring. Detailed monitoring does incur charges. For more information, see AWS
  CloudWatch documentation.",
           "type": "array",
           "items": {
             "type": "string",
             "enum": [
               "enabled",
               "disabled"
             ]
           }
         }
       },
       "metadata": {
         "ui:order": [
           "InstanceIds",
           "MonitoringValue"
         ]
       },
       "additionalProperties": false,
       "required": [
         "InstanceIds",
         "MonitoringValue"
       ]
     }
   },
   "metadata": {
     "ui:order": [
       "DocumentName",
       "Region",
       "Parameters"
     ]
   },
   "additionalProperties": false,
   "required": [
     "DocumentName",
     "Region",
     "Parameters"
   ]
}
```

Schema for Change Type ct-0tpbr6lfa3zng

Classifications:

- Management | Advanced stack components | Application Load Balancer | Remove listener certificate


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove ALB Listener Certificate",
  "description": "Remove a certificate from the specified Application Load Balancer
 (ALB) listener. Use the RemediateStackDrift parameter for the automation to try to
 remediate drift, if it is introduced.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RemoveCertificateFromElbv2Listener.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveCertificateFromElbv2Listener"
      ],
      "default": "AWSManagedServices-RemoveCertificateFromElbv2Listener"
    },
    "Region": {
      "description": "The AWS Region where the application load balancer listener is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ListenerArn": {
          "description": "The Amazon Resource Name (ARN) of the listener in
 the form arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
sample/1234567890abcdfe/1234567890abcdfe.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):elasticloadbalancing:[a-z]{2}-[a-
z]+-[0-9]{1}:[0-9]{12}:listener/[a-z]{3}/[A-Za-z0-9-]+/[a-z0-9-]+/[a-z0-9-]+$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        },
        "CertificateArn": {
          "description": "The Amazon Resource Name (ARN) of the certificate in the form
 arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused by
 removing the certificate from the Loadbalancer Listener. False to not attempt drift
 remediation. Drift remediation can be performed only on CloudFormation stacks that
 were created using a CT other than the Ingestion CT ct-36cn2avfrrj9v and that are
 in sync with the definitions in the stack template prior to removing the certificate
 from the Loadbalancer Listener. Set to False to remove the certificate from the
 Loadbalancer Listener in an ingested stack if any drift introduced by the change is
 acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ListenerArn",
          "CertificateArn",
          "RemediateStackDrift"
        ]
```

```
      },
      "additionalProperties": false,
      "required": [
        "CertificateArn",
        "ListenerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0ttx8eh3ice91

Classifications:

- Management | Advanced stack components | S3 storage | Delete policy (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete policy",
  "description": "Use to delete an S3 bucket policy.",
  "type": "object",
  "properties": {
    "BucketName": {
      "description": "S3 Bucket to delete the bucket policy from.",
      "type": "string",
      "pattern": "^[A-Za-z0-9][A-Za-z0-9\\-]{1,61}[A-Za-z0-9]$",
      "maxLength": 63
    },
    "Operation": {
```

```
      "description": "Must be Delete policy.",
      "type": "string",
      "default": "Delete policy",
      "enum": [
        "Delete policy"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "BucketName",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "BucketName",
    "Operation"
  ]
}
```

Schema for Change Type ct-0vdiy51oyrhhm

Classifications:

- Management | Managed landing zone | Management account | Offboard application account

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Offboard Application Account",
```

```
  "description": "Offboard the specified application account. Run this from the
 management account for the application account that you want offboarded. You must
 first confirm the offboarding request by submitting the Confirm offboarding CT
 (ct-2wlfo2jxj2rkj) from the application account. If you are offboarding a customer-
managed account, then ct-2wlfo2jxj2rkj is not needed. Only use these CTs when you
 plan to terminate all resources within the specified account and close the account.
 After you successfully submit both CTs, AMS can't undo the offboarding, repurpose the
 account, or help you to remediate issues in the account.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be OffboardingExecution.",
      "type": "string",
      "enum": [
        "OffboardingExecution"
      ],
      "default": "OffboardingExecution"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AccountId": {
          "description": "The unique identifier (ID) of the application account to
 offboard.",
          "type": "string",
          "pattern": "^[0-9]{12}$"
        },
        "AccountEmail": {
          "description": "The email associated with the application account to
 offboard.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
        },
        "Confirmation": {
          "description": "To offboard the provided application account, confirm the
 operation by specifying 'confirm' in the text input field.",
          "type": "string",
          "pattern": "confirm"
        },
        "DeleteTransitGatewayAttachment": {
          "description": "Specify true to delete the attachment to the default Transit
 Gateway within core networking account. Set to false to retain the connectivity using
 Transit Gateway.",
          "type": "boolean"
```

```
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AccountId",
          "AccountEmail",
          "Confirmation",
          "DeleteTransitGatewayAttachment"
        ]
      },
      "required": [
        "AccountId",
        "AccountEmail",
        "Confirmation",
        "DeleteTransitGatewayAttachment"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Parameters",
      "RequestType"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Parameters",
    "RequestType"
  ]
}
```

Schema for Change Type ct-0vevjppj9eta4

Classifications:

- Management | Advanced stack components | EBS Volume | Encrypt EBS by default

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Encrypt EBS By Default",
```

```
  "description": "Set Amazon Elastic Block Store (EBS) to enforce the encryption.
 After you enable encryption by default, the EBS volumes that you create and snapshot
 copies are always encrypted, either using the KMS key configured as default for EBS
 encryption or the key that you specified when you created each volume.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-EncryptEBSByDefault.",
      "type": "string",
      "enum": [
        "AWSManagedServices-EncryptEBSByDefault"
      ],
      "default": "AWSManagedServices-EncryptEBSByDefault"
    },
    "Region": {
      "description": "The AWS Region to enable EBS encryption by default in, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region"
    ]
  },
  "required": [
    "DocumentName",
    "Region"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0vzsr2nyraedl

Classifications:

- Deployment | Advanced stack components | DNS (public) | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
 "name": "Create Public DNS Record",
 "description": "Create a new Route 53 DNS resource record set and a new public hosted
zone for a VPC, and configure traffic routing.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-CreateAddRoute53Resources.",
     "type": "string",
     "enum": [
       "AWSManagedServices-CreateAddRoute53Resources"
     ],
     "default": "AWSManagedServices-CreateAddRoute53Resources"
   },
   "Region": {
     "description": "The AWS Region in which the AWS resource is located, in the form
us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "description": "Specifications for the stack.",
     "type": "object",
     "properties": {
       "DomainName": {
         "description": "A domain name for the hosted zone. The name can contain only
lowercase letters, numbers, hyphens (-), and a dot (.). For example, mycorp.com",
         "type": "string",
         "minLength": 2,
         "pattern": "^([a-z0-9]+(-[a-z0-9]+)*\\.)+[a-z]{2,255}$"
       },
       "DomainType": {
         "description": "Must be 'public'",
         "type": "string",
         "enum": [
           "public"
         ],
         "default": "public"
       },
       "RecordSet": {
         "description": "A JSON of resource records for the hosted zone.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^\\s*\\{\\s*\"RecordSet\"\\s*:\\s*\\[.*\\]\\s*\\}\\s*$"
```

```
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "DomainName",
          "DomainType",
          "RecordSet"
        ]
      },
      "required": [
        "DomainName",
        "DomainType",
        "RecordSet"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0wglhholzo0uw

Classifications:

- Management | Advanced stack components | Network Load Balancer | Update

```
{
```

```
 "$schema": "http://json-schema.org/draft-04/schema#",
 "name": "Update Network Load Balancer",
 "description": "Update the properties of an existing Network Load Balancer.",
 "type": "object",
 "properties": {
   "VpcId": {
     "description": "The ID of the VPC where the Network Load Balancer is, in the form
vpc-0123abcd or vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "StackId": {
     "description": "The stack ID of the Network Load Balancer that you are updating,
in the form stack-a1b2c3d4e5f67890e.",
     "type": "string",
     "pattern": "^stack-[a-z0-9]{17}$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "HealthCheckHealthyThreshold": {
         "type": "string",
         "description": "The number of consecutive health check successes required to
declare an EC2 instance healthy.",
         "pattern": "[2-9]{1}|10"
       },
       "HealthCheckIntervalSeconds": {
         "type": "string",
         "description": "The approximate interval, in seconds, between health
checks.",
         "enum": [
           "10",
           "30"
         ]
       },
       "HealthCheckTargetPath": {
         "type": "string",
         "description": "The ping path destination on the application hosts
where the load balancer sends health check requests. This is only applicable if
HealthCheckTargetProtocol = HTTP or HTTPS."
       },
       "HealthCheckTargetPort": {
         "type": "string",
```

```
        "description": "The port the load balancer uses when performing health checks
on targets. The default is traffic-port, which indicates the port on which each target
receives traffic from the load balancer.",
        "pattern": "([0-9]{1,5})?"
      },
      "HealthCheckTargetProtocol": {
        "type": "string",
        "description": "The protocol the load balancer uses when performing health
checks on targets.",
        "enum": [
          "HTTP",
          "HTTPS",
          "TCP"
        ]
      },
      "CrossZoneEnabled": {
        "type": "string",
        "description": "True if cross-zone load balancing is enabled. False if it is
not.",
        "enum": [
          "true",
          "false"
        ]
      },
      "SubnetIds": {
        "type": "array",
        "description": "One or more subnet IDs for the load balancer, in the form
subnet-0123abcd or subnet-01234567890abcdef. Please note that if you update SubnetIds,
the new value must contain all of the required SubnetIds for the NLB, the new ones and
the ones used before.",
        "items": {
          "type": "string"
        }
      },
      "ProxyProtocolV2": {
        "type": "string",
        "description": "True if proxy protocol version 2 is enabled. False if it is
not.",
        "enum": [
          "true",
          "false"
        ]
      },
      "DeregistrationDelayTimeoutSeconds": {
```

```
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing to
wait before changing the state of a deregistering target from draining to unused.",
          "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})"
        },
        "Target1ID": {
          "type": "string",
          "description": "The ID of the EC2 instance to register a target if the
TargetType = instance, in the form i-0123abcd or i-01234567890abcdef. Leave blank if
you don't need to register a target."
        },
        "Target1Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic."
        },
        "Target1AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target1ID is
outside the VPC. Leave blank if TargetType = instance."
        },
        "Target2ID": {
          "type": "string",
          "description": "The ID of the EC2 instance to register a target if the
TargetType = instance, in the form i-0123abcd or i-01234567890abcdef. Leave blank if
you don't need to register a target."
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic."
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target2ID is
outside the VPC. Leave blank if TargetType = instance."
        },
        "Target3ID": {
```

```
          "type": "string",
          "description": "The ID of the EC2 instance to register a target if the
TargetType = instance, in the form i-0123abcd or i-01234567890abcdef. Leave blank if
you don't need to register a target."
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic."
        },
        "Target3AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target3ID is
outside the VPC. Leave blank if TargetType = instance."
        },
        "Target4ID": {
          "type": "string",
          "description": "The ID of the EC2 instance to register a target if the
TargetType = instance, in the form i-0123abcd or i-01234567890abcdef. Leave blank if
you don't need to register a target."
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic."
        },
        "Target4AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target4ID is
outside the VPC. Leave blank if TargetType = instance."
        }
      },
      "metadata": {
        "ui:order": [
          "ProxyProtocolV2",
          "DeregistrationDelayTimeoutSeconds",
          "CrossZoneEnabled",
          "SubnetIds",
```

```
          "HealthCheckTargetPath",
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckIntervalSeconds",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
          "Target4AvailabilityZone"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0wspy4o646g9p

Classifications:

- Management | Host security | Trend Micro DSM | Add login (read-only)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add DSM Read-Only Login",
  "description": "Request a read-only login to the Trend Micro console for your
 account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateEPSDSMReadOnlyUser.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateEPSDSMReadOnlyUser"
      ],
      "default": "AWSManagedServices-CreateEPSDSMReadOnlyUser"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}|^$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Username": {
          "description": "The username for the EPS user. The name can be up to 50
 characters in length.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9._\\-:/()#,@\\[\\]+=&;{}!$\\*]{1,50}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "FullName": {
          "description": "The full name for the EPS user. The name can be up to 50
 characters in length.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^[ a-zA-Z0-9._\\-:/()#,@\\[\\]+=&;{}!$\\*]{1,50}$"
          },
          "minItems": 0,
```

```
          "maxItems": 1
        },
        "Description": {
          "description": "The description for the EPS user. The description can be up
 to 150 characters in length.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^[ a-zA-Z0-9._\\-:/()#,@\\[\\]+=&;{}!$\\*]{1,150}$"
          },
          "minItems": 0,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Username",
          "FullName",
          "Description"
        ]
      },
      "additionalProperties": false,
      "required": [
        "Username"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0x6dylrnfjgz5

Classifications:

- Management | Directory Service | Directory | Create AD trust


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Active Directory Trust",
  "description": "Create a one-way trust between On-Prem Domain and (AWS) Managed
 Active Directory. For multi-account landing zone (MALZ), use this change type in the
 shared services account. Before creating the trust, you need to make sure that the
 following prerequisites are met: 1. You must create the AD trust first on the On-
Prem Domain and save the trust password in the Secrets Manager. 2. You must set up a
 Managed Active Directory (MAD) Security Group with an outbound rule that allows all
 traffic to On-Prem CIDR ranges.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateADTrust.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateADTrust"
      ],
      "default": "AWSManagedServices-CreateADTrust"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
```

```
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DirectoryId": {
          "description": "The Directory ID of the Managed Microsoft AD directory for
 which to establish the trust relationship.",
          "type": "string",
          "pattern": "^d-[0-9a-f]{10}$"
        },
        "RemoteDomainName": {
          "description": "The Fully Qualified Domain Name (FQDN) of the external domain
 for which to create the trust relationship.",
          "type": "string",
          "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+[.]?$"
        },
        "SecretArn": {
          "description": "ARN of the secret where the AD trust password is stored. The
 secret must be stored as a string value not as a key/value pair. The secret name must
 be prefixed with customer-shared/; for example, customer-shared/trustpassword.",
          "type": "string",
          "pattern": "arn:(aws|aws-cn|aws-us-gov):secretsmanager:[a-z]{2}-[a-z]+-[0-9]
{1}:\\d{12}:secret:([cC][uU][sS][tT][oO][mM][eE][rR]-[sS][hH][aA][rR][eE][dD])[\\w/_
+=.@-]{1,512}"
        },
        "TrustType": {
          "description": "The trust relationship type.",
          "type": "string",
          "enum": [
```

```
            "Forest",
            "External"
          ]
        },
        "ConditionalForwarderIpAddresses": {
          "description": "A comma-delimited list of one or more IP addresses of the
 remote DNS server associated with RemoteDomainName.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(,(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|
25[0-5])\\.){3}([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]))*$"
        }
      },
      "metadata": {
        "ui:order": [
          "DirectoryId",
          "RemoteDomainName",
          "SecretArn",
          "TrustType",
          "ConditionalForwarderIpAddresses"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DirectoryId",
        "RemoteDomainName",
        "SecretArn",
        "TrustType",
        "ConditionalForwarderIpAddresses"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-0xdawir96cy7k

Classifications:

- Management | Other | Other | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update other",
  "description": "Use to request a manual update to a resource.",
  "type": "object",
  "properties": {
    "Comment": {
      "description": "The description of the change.",
      "type": "string",
      "maxLength": 5000
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    },
    "RelatedIds": {
      "description": "(Optional) IDs of resources related to the change request.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 1000,
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
```

```
  "required": [
    "Comment"
  ],
  "metadata": {
    "ui:order": [
      "Comment",
      "RelatedIds",
      "Priority"
    ]
  }
}
```

Schema for Change Type ct-0xi6q7uwuwrqe

Classifications:

- Deployment | Advanced stack components | Cache (ElastiCache Memcached) stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Cache (ElastiCache Memcached) stack",
  "description": "Use to create an Amazon ElastiCache cluster (one or more cache nodes)
 that uses the Memcached engine, and specify CloudWatch metrics and alarms for the
 cluster.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the vpc to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-sfpo2o00000000000.",
      "type": "string",
      "enum": [
```

```
        "stm-sfpo2o00000000000"
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
```

```
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "ElastiCacheAutoMinorVersionUpgrade": {
          "description": "True for minor engine upgrades to be applied automatically
to the cache cluster during the specified ElastiCachePreferredMaintenanceWindow, false
for the upgrades to not be applied automatically. Default is true.",
          "type": "boolean",
          "default": true
        },
        "ElastiCacheAvailabilityZones": {
          "description": "One or more Availability Zones where cache nodes will be
created.",
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 1
        },
        "ElastiCacheClusterName": {
          "description": "A name for the cache cluster.",
          "type": "string",
          "minLength": 1,
          "maxLength": 20,
          "pattern": "^[a-zA-Z][a-zA-Z0-9-]{0,18}[a-zA-Z0-9]$|^[a-zA-Z]$"
        },
        "ElastiCacheCPUThresholdAlarmOverride": {
          "description": "The optional value for the CPUUtilization metric maximum
threshold to use instead of the default value for the instance type.",
          "type": "number",
          "default": 0,
          "minimum": 0,
          "maximum": 100
        },
        "ElastiCacheEngine": {
          "description": "Must be memcached.",
          "type": "string",
          "enum": [
            "memcached"
          ]
```

```
        },
        "ElastiCacheEngineVersion": {
          "description": "The version of the Memcached engine to be used for this
 cluster.",
          "type": "string"
        },
        "ElastiCacheInstanceType": {
          "description": "The compute and memory capacity of nodes in the cache
 cluster.",
          "type": "string",
          "default": "cache.t3.micro"
        },
        "ElastiCacheMultiAZ": {
          "description": "True for the nodes to be created in a single Availability
 Zone, false for them to be created across multiple Availability Zones in the cluster's
 region. Default is false.",
          "type": "boolean",
          "default": false
        },
        "ElastiCacheNumberOfNodes": {
          "description": "The number of cache nodes that the Memcached cluster should
 have.",
          "type": "number",
          "default": 1,
          "minimum": 1,
          "maximum": 20
        },
        "ElastiCachePort": {
          "description": "The port number on which each of the cache nodes will accept
 connections.",
          "type": "number",
          "minimum": 0,
          "maximum": 65535,
          "default": 11211
        },
        "ElastiCachePreferredMaintenanceWindow": {
          "description": "The weekly time range (in UTC) during which system
 maintenance can occur. For example, you can specify: sun:02:00-sun:04:00.",
          "type": "string",
          "pattern": "^(?:sun|mon|tue|wed|thu|fri|sat):(?:[0-1][0-9]|2[0-3]):[0-5]
[0-9]-(?:sun|mon|tue|wed|thu|fri|sat):(?:[0-1][0-9]|2[0-3]):[0-5][0-9]$"
        },
        "ElastiCacheSubnetGroup": {
```

```
          "description": "The name of the subnet group to associate with the
cluster.",
          "type": "string",
          "minLength": 1,
          "maxLength": 255,
          "pattern": "^[a-z0-9-]{1,255}$"
        },
        "ElastiCacheSubnetIds": {
          "description": "One or more subnet IDs for the cache cluster, in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1
        },
        "SecurityGroups": {
          "description": "One or more VPC security groups to associate with the
cluster, in the form sg-0123abcd or sg-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 1
        }
      },
      "additionalProperties": false,
      "required": [
        "ElastiCacheAvailabilityZones",
        "ElastiCacheClusterName",
        "ElastiCacheEngine",
        "ElastiCacheSubnetIds"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "Parameters",
```

```
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-0xqwmtn1hfh8u

Classifications:

- Management | Advanced stack components | Tag | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Resource Tags",
  "description": "Update tags on existing, tagged resources: Autoscaling, EC2,
 Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally,
 CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS
 infrastructure stacks (stacks named mc-*) cannot have tags updated with this change
 type.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateTags.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateTags"
      ],
      "default": "AWSManagedServices-UpdateTags"
    },
    "Region": {
      "description": "The AWS Region where the tagged resources are, in the form us-
east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ResourceArns": {
          "description": "A list of up to 50 Amazon resource names (ARNs), or the
 resource IDs, of the resources with tags to be updated. Use resource ID only for these
 resource types: EC2 instance, EBS volume, EBS snapshot, AMI, and security group. Use
 the full ARN for all other supported resource types.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:(autoscaling|ec2|elasticloadbalancing|logs|rds|s3|
redshift):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}):(|[0-9]{12}):.*)$|^(ami|i|vol|
sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "AddOrUpdateTags": {
          "description": "Up to fifty tags (key/value pairs) to categorize the
 resource, in the form {\"Key\":\"TagKey1\",\"Value\":\"TagValue1\"}. If the tag
 exists, the value for it is overwritten. If the tag does not exist, it is added to the
 resource. Characters allowed in tags can vary by AWS service. For information about
 what characters can be used to tag resources in a particular AWS service, please refer
 to its documentation. In general, allowed characters in tags are letters, numbers,
 spaces and the following characters: _ . : / = + - @.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\{\\}$|^\\{\"Key\":\"((aws-migration-project-id)|(?![aA]
[mMwW][sS])[\\x00-\\x7F+]{1,128})\",\"Value\":\"[\\x00-\\x7F+]{0,255}\"\\}"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "RemoveTags": {
          "description": "Up to fifty tag Keys to remove from the specified
 resource.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((aws-migration-project-id)|(?![aA][mMwW][sS])[\\x00-\\x7F+]
{1,128})$",
            "minLength": 1,
            "maxLength": 127
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "ResourceArns",
          "AddOrUpdateTags",
          "RemoveTags"
        ]
      },
      "required": [
        "ResourceArns"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-0ywnhc8e5k9z5

Classifications:

- Deployment | AMS Resource Scheduler | Solution | Deploy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Deploy AMS Resource Scheduler",
  "description": "Deploy the AMS Resource Scheduler solution in the account. The AMS
 Resource Scheduler lets you schedule automatic start and/or stop for Auto Scaling
 groups, EC2s, and RDS instances. Note that the Resource Scheduler deploys in an
```

```
 enabled state, by default; you can manage that with the AMS Resource Scheduler Disable
 and Enable change types.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAMSResourceSchedulerStack-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
      ],
      "default": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account for the AMS Resource Scheduler
 solution to be deployed, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SchedulingActive": {
          "description": "Yes to enable the Resource Scheduler. No to disable it.
 The default is Yes. Use Resource Scheduler enable (ct-2wrvu4kca9xky) and disable
 (ct-14v49adibs4db) change types to manage state.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Yes",
              "No"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ScheduledServices": {
          "description": "Comma-separated list of scheduled services. Use a combination
 of AutoScaling, EC2, and RDS.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|(^(ec2|rds|autoscaling)(,(ec2|rds|autoscaling)){0,2}$)"
```

```
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TagName": {
          "description": "The name of the tag key to use to associate the instance
 schedule schemas with service resources. Default is Schedule.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^(?!(aws:|ams:))[a-zA-Z0-9+-=._:/@]{1,127}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "UseCMK": {
          "description": "Comma-separated list of Customer Managed Key (CMK) Amazon
 Resource Names (ARNs) in format arn:<partition>:kms:<region>:<account-id>:key/<key-id>
 to grant Resource Scheduler permission to. These are CMK that are used to encrypt EBS
 volumes on EC2 instances.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(|arn:(aws|aws-cn|aws-us-gov):kms:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{0,12}:key/[a-z0-9\\-]+)$"
          },
          "minItems": 1,
          "maxItems": 20
        },
        "UseLicenseManager": {
          "description": "Comma-separated list of AWS License Manager license ARNs to
 grant Resource Scheduler permission to. These are software or vendor licenses that EC2
 instances are configured with.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(|arn:(aws|aws-cn|aws-us-gov):license-manager:([a-z]{2}((-
gov))?-[a-z]+-\\d{1}):[0-9]{0,12}:license-configuration(/|:)lic-.*)$"
          },
          "minItems": 1,
          "maxItems": 20
        },
        "DefaultTimezone": {
```

```
          "description": "The name of the timezone, in the form US/Pacific, to be used
 as the default timezone. The default is UTC.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Africa/Abidjan",
              "Africa/Accra",
              "Africa/Addis_Ababa",
              "Africa/Algiers",
              "Africa/Asmara",
              "Africa/Bamako",
              "Africa/Bangui",
              "Africa/Banjul",
              "Africa/Bissau",
              "Africa/Blantyre",
              "Africa/Brazzaville",
              "Africa/Bujumbura",
              "Africa/Cairo",
              "Africa/Casablanca",
              "Africa/Ceuta",
              "Africa/Conakry",
              "Africa/Dakar",
              "Africa/Dar_es_Salaam",
              "Africa/Djibouti",
              "Africa/Douala",
              "Africa/El_Aaiun",
              "Africa/Freetown",
              "Africa/Gaborone",
              "Africa/Harare",
              "Africa/Johannesburg",
              "Africa/Juba",
              "Africa/Kampala",
              "Africa/Khartoum",
              "Africa/Kigali",
              "Africa/Kinshasa",
              "Africa/Lagos",
              "Africa/Libreville",
              "Africa/Lome",
              "Africa/Luanda",
              "Africa/Lubumbashi",
              "Africa/Lusaka",
              "Africa/Malabo",
              "Africa/Maputo",
```

```
                "Africa/Maseru",
                "Africa/Mbabane",
                "Africa/Mogadishu",
                "Africa/Monrovia",
                "Africa/Nairobi",
                "Africa/Ndjamena",
                "Africa/Niamey",
                "Africa/Nouakchott",
                "Africa/Ouagadougou",
                "Africa/Porto-Novo",
                "Africa/Sao_Tome",
                "Africa/Tripoli",
                "Africa/Tunis",
                "Africa/Windhoek",
                "America/Adak",
                "America/Anchorage",
                "America/Anguilla",
                "America/Antigua",
                "America/Araguaina",
                "America/Argentina/Buenos_Aires",
                "America/Argentina/Catamarca",
                "America/Argentina/Cordoba",
                "America/Argentina/Jujuy",
                "America/Argentina/La_Rioja",
                "America/Argentina/Mendoza",
                "America/Argentina/Rio_Gallegos",
                "America/Argentina/Salta",
                "America/Argentina/San_Juan",
                "America/Argentina/San_Luis",
                "America/Argentina/Tucuman",
                "America/Argentina/Ushuaia",
                "America/Aruba",
                "America/Asuncion",
                "America/Atikokan",
                "America/Bahia",
                "America/Bahia_Banderas",
                "America/Barbados",
                "America/Belem",
                "America/Belize",
                "America/Blanc-Sablon",
                "America/Boa_Vista",
                "America/Bogota",
                "America/Boise",
                "America/Cambridge_Bay",
```

```
                "America/Campo_Grande",
                "America/Cancun",
                "America/Caracas",
                "America/Cayenne",
                "America/Cayman",
                "America/Chicago",
                "America/Chihuahua",
                "America/Costa_Rica",
                "America/Creston",
                "America/Cuiaba",
                "America/Curacao",
                "America/Danmarkshavn",
                "America/Dawson",
                "America/Dawson_Creek",
                "America/Denver",
                "America/Detroit",
                "America/Dominica",
                "America/Edmonton",
                "America/Eirunepe",
                "America/El_Salvador",
                "America/Fortaleza",
                "America/Glace_Bay",
                "America/Godthab",
                "America/Goose_Bay",
                "America/Grand_Turk",
                "America/Grenada",
                "America/Guadeloupe",
                "America/Guatemala",
                "America/Guayaquil",
                "America/Guyana",
                "America/Halifax",
                "America/Havana",
                "America/Hermosillo",
                "America/Indiana/Indianapolis",
                "America/Indiana/Knox",
                "America/Indiana/Marengo",
                "America/Indiana/Petersburg",
                "America/Indiana/Tell_City",
                "America/Indiana/Vevay",
                "America/Indiana/Vincennes",
                "America/Indiana/Winamac",
                "America/Inuvik",
                "America/Iqaluit",
                "America/Jamaica",
```

```
                "America/Juneau",
                "America/Kentucky/Louisville",
                "America/Kentucky/Monticello",
                "America/Kralendijk",
                "America/La_Paz",
                "America/Lima",
                "America/Los_Angeles",
                "America/Lower_Princes",
                "America/Maceio",
                "America/Managua",
                "America/Manaus",
                "America/Marigot",
                "America/Martinique",
                "America/Matamoros",
                "America/Mazatlan",
                "America/Menominee",
                "America/Merida",
                "America/Metlakatla",
                "America/Mexico_City",
                "America/Miquelon",
                "America/Moncton",
                "America/Monterrey",
                "America/Montevideo",
                "America/Montreal",
                "America/Montserrat",
                "America/Nassau",
                "America/New_York",
                "America/Nipigon",
                "America/Nome",
                "America/Noronha",
                "America/North_Dakota/Beulah",
                "America/North_Dakota/Center",
                "America/North_Dakota/New_Salem",
                "America/Ojinaga",
                "America/Panama",
                "America/Pangnirtung",
                "America/Paramaribo",
                "America/Phoenix",
                "America/Port-au-Prince",
                "America/Port_of_Spain",
                "America/Porto_Velho",
                "America/Puerto_Rico",
                "America/Rainy_River",
                "America/Rankin_Inlet",
```

```
                "America/Recife",
                "America/Regina",
                "America/Resolute",
                "America/Rio_Branco",
                "America/Santa_Isabel",
                "America/Santarem",
                "America/Santiago",
                "America/Santo_Domingo",
                "America/Sao_Paulo",
                "America/Scoresbysund",
                "America/Sitka",
                "America/St_Barthelemy",
                "America/St_Johns",
                "America/St_Kitts",
                "America/St_Lucia",
                "America/St_Thomas",
                "America/St_Vincent",
                "America/Swift_Current",
                "America/Tegucigalpa",
                "America/Thule",
                "America/Thunder_Bay",
                "America/Tijuana",
                "America/Toronto",
                "America/Tortola",
                "America/Vancouver",
                "America/Whitehorse",
                "America/Winnipeg",
                "America/Yakutat",
                "America/Yellowknife",
                "Antarctica/Casey",
                "Antarctica/Davis",
                "Antarctica/DumontDUrville",
                "Antarctica/Macquarie",
                "Antarctica/Mawson",
                "Antarctica/McMurdo",
                "Antarctica/Palmer",
                "Antarctica/Rothera",
                "Antarctica/Syowa",
                "Antarctica/Vostok",
                "Arctic/Longyearbyen",
                "Asia/Aden",
                "Asia/Almaty",
                "Asia/Amman",
                "Asia/Anadyr",
```

```
                "Asia/Aqtau",
                "Asia/Aqtobe",
                "Asia/Ashgabat",
                "Asia/Baghdad",
                "Asia/Bahrain",
                "Asia/Baku",
                "Asia/Bangkok",
                "Asia/Beirut",
                "Asia/Bishkek",
                "Asia/Brunei",
                "Asia/Choibalsan",
                "Asia/Chongqing",
                "Asia/Colombo",
                "Asia/Damascus",
                "Asia/Dhaka",
                "Asia/Dili",
                "Asia/Dubai",
                "Asia/Dushanbe",
                "Asia/Gaza",
                "Asia/Harbin",
                "Asia/Hebron",
                "Asia/Ho_Chi_Minh",
                "Asia/Hong_Kong",
                "Asia/Hovd",
                "Asia/Irkutsk",
                "Asia/Jakarta",
                "Asia/Jayapura",
                "Asia/Jerusalem",
                "Asia/Kabul",
                "Asia/Kamchatka",
                "Asia/Karachi",
                "Asia/Kashgar",
                "Asia/Kathmandu",
                "Asia/Khandyga",
                "Asia/Kolkata",
                "Asia/Krasnoyarsk",
                "Asia/Kuala_Lumpur",
                "Asia/Kuching",
                "Asia/Kuwait",
                "Asia/Macau",
                "Asia/Magadan",
                "Asia/Makassar",
                "Asia/Manila",
                "Asia/Muscat",
```

```
                "Asia/Nicosia",
                "Asia/Novokuznetsk",
                "Asia/Novosibirsk",
                "Asia/Omsk",
                "Asia/Oral",
                "Asia/Phnom_Penh",
                "Asia/Pontianak",
                "Asia/Pyongyang",
                "Asia/Qatar",
                "Asia/Qyzylorda",
                "Asia/Rangoon",
                "Asia/Riyadh",
                "Asia/Sakhalin",
                "Asia/Samarkand",
                "Asia/Seoul",
                "Asia/Shanghai",
                "Asia/Singapore",
                "Asia/Taipei",
                "Asia/Tashkent",
                "Asia/Tbilisi",
                "Asia/Tehran",
                "Asia/Thimphu",
                "Asia/Tokyo",
                "Asia/Ulaanbaatar",
                "Asia/Urumqi",
                "Asia/Ust-Nera",
                "Asia/Vientiane",
                "Asia/Vladivostok",
                "Asia/Yakutsk",
                "Asia/Yekaterinburg",
                "Asia/Yerevan",
                "Atlantic/Azores",
                "Atlantic/Bermuda",
                "Atlantic/Canary",
                "Atlantic/Cape_Verde",
                "Atlantic/Faroe",
                "Atlantic/Madeira",
                "Atlantic/Reykjavik",
                "Atlantic/South_Georgia",
                "Atlantic/St_Helena",
                "Atlantic/Stanley",
                "Australia/Adelaide",
                "Australia/Brisbane",
                "Australia/Broken_Hill",
```

```
                "Australia/Currie",
                "Australia/Darwin",
                "Australia/Eucla",
                "Australia/Hobart",
                "Australia/Lindeman",
                "Australia/Lord_Howe",
                "Australia/Melbourne",
                "Australia/Perth",
                "Australia/Sydney",
                "Canada/Atlantic",
                "Canada/Central",
                "Canada/Eastern",
                "Canada/Mountain",
                "Canada/Newfoundland",
                "Canada/Pacific",
                "Europe/Amsterdam",
                "Europe/Andorra",
                "Europe/Athens",
                "Europe/Belgrade",
                "Europe/Berlin",
                "Europe/Bratislava",
                "Europe/Brussels",
                "Europe/Bucharest",
                "Europe/Budapest",
                "Europe/Busingen",
                "Europe/Chisinau",
                "Europe/Copenhagen",
                "Europe/Dublin",
                "Europe/Gibraltar",
                "Europe/Guernsey",
                "Europe/Helsinki",
                "Europe/Isle_of_Man",
                "Europe/Istanbul",
                "Europe/Jersey",
                "Europe/Kaliningrad",
                "Europe/Kiev",
                "Europe/Lisbon",
                "Europe/Ljubljana",
                "Europe/London",
                "Europe/Luxembourg",
                "Europe/Madrid",
                "Europe/Malta",
                "Europe/Mariehamn",
                "Europe/Minsk",
```

```
                "Europe/Monaco",
                "Europe/Moscow",
                "Europe/Oslo",
                "Europe/Paris",
                "Europe/Podgorica",
                "Europe/Prague",
                "Europe/Riga",
                "Europe/Rome",
                "Europe/Samara",
                "Europe/San_Marino",
                "Europe/Sarajevo",
                "Europe/Simferopol",
                "Europe/Skopje",
                "Europe/Sofia",
                "Europe/Stockholm",
                "Europe/Tallinn",
                "Europe/Tirane",
                "Europe/Uzhgorod",
                "Europe/Vaduz",
                "Europe/Vatican",
                "Europe/Vienna",
                "Europe/Vilnius",
                "Europe/Volgograd",
                "Europe/Warsaw",
                "Europe/Zagreb",
                "Europe/Zaporozhye",
                "Europe/Zurich",
                "GMT",
                "Indian/Antananarivo",
                "Indian/Chagos",
                "Indian/Christmas",
                "Indian/Cocos",
                "Indian/Comoro",
                "Indian/Kerguelen",
                "Indian/Mahe",
                "Indian/Maldives",
                "Indian/Mauritius",
                "Indian/Mayotte",
                "Indian/Reunion",
                "Pacific/Apia",
                "Pacific/Auckland",
                "Pacific/Chatham",
                "Pacific/Chuuk",
                "Pacific/Easter",
```

```
            "Pacific/Efate",
            "Pacific/Enderbury",
            "Pacific/Fakaofo",
            "Pacific/Fiji",
            "Pacific/Funafuti",
            "Pacific/Galapagos",
            "Pacific/Gambier",
            "Pacific/Guadalcanal",
            "Pacific/Guam",
            "Pacific/Honolulu",
            "Pacific/Johnston",
            "Pacific/Kiritimati",
            "Pacific/Kosrae",
            "Pacific/Kwajalein",
            "Pacific/Majuro",
            "Pacific/Marquesas",
            "Pacific/Midway",
            "Pacific/Nauru",
            "Pacific/Niue",
            "Pacific/Norfolk",
            "Pacific/Noumea",
            "Pacific/Pago_Pago",
            "Pacific/Palau",
            "Pacific/Pitcairn",
            "Pacific/Pohnpei",
            "Pacific/Port_Moresby",
            "Pacific/Rarotonga",
            "Pacific/Saipan",
            "Pacific/Tahiti",
            "Pacific/Tarawa",
            "Pacific/Tongatapu",
            "Pacific/Wake",
            "Pacific/Wallis",
            "US/Alaska",
            "US/Arizona",
            "US/Central",
            "US/Eastern",
            "US/Hawaii",
            "US/Mountain",
            "US/Pacific",
            "UTC"
          ]
        },
        "minItems": 1,
```

```
          "maxItems": 1
        },
        "Action": {
          "description": "Must be Deploy.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Deploy"
            ],
            "default": "Deploy"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SchedulingActive",
          "ScheduledServices",
          "TagName",
          "DefaultTimezone",
          "UseCMK",
          "UseLicenseManager",
          "Action"
        ]
      },
      "required": [
        "Action"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-0zko7t3rk2efb

Classifications:

- Management | Advanced stack components | Tag | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Resource Tags (Review Required)",
  "description": "Add tags to, update tags on, or remove tags from, existing,
 supported, resources except those in AMS infrastructure stacks (stacks named mc-
*). Tags simplify categorization, identification and targeting AWS resources. Use
 BulkUpdate if you have >50 tags to manage. For Autoscaling, EC2, Elastic Load
 Balancing, RDS resources and S3 buckets, use automated CT ct-0xqwmtn1hfh8u.",
  "type": "object",
  "properties": {
    "Resources": {
      "description": "Parameters for up to fifty resources for tag management.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "ResourceArn": {
            "description": "The ARN or the resource ID of the resource to be tagged.
 Resource ID is allowed only for these resource types: EC2 instance, EBS volume, EBS
 snapshot, AMI, and security group. All other resource types must be provided with the
 full ARN.",
            "type": "string",
            "pattern": "^arn:aws:(|[a-z][a-z0-9-]+):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-
z]+-\\d{1}):(|[0-9]{12}):([^,\\s]+)$|^(ami|i|vol|sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "AddOrUpdateTags": {
            "description": "Up to fifty tags (key/value pairs) to add to, or update
 for, the specified resources. If the tag exists, the value for it is overwritten.
 If the tag does not exist, it is added to the resource. Characters allowed in tags
 can vary by AWS service. For information about what characters can be used to tag
 resources in a particular AWS service, please refer to its documentation. In general,
```

```
allowed characters in tags are letters, numbers, spaces and the following characters:
_ . : / = + - @.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "Key": {
                "type": "string",
                "pattern": "^(?![aA][mMwW][sS]:)[a-zA-Z0-9\\s_.:/=+\\\\\\\-@\\]*]+$",
                "minLength": 1,
                "maxLength": 127
              },
              "Value": {
                "type": "string",
                "pattern": "^[a-zA-Z0-9\\s_.:/=+\\\\\\\-@\\]*]+$",
                "minLength": 1,
                "maxLength": 255
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "Key",
                "Value"
              ]
            },
            "required": [
              "Key",
              "Value"
            ]
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "RemoveTags": {
          "description": "Up to fifty tag Keys to remove from the specified
resource.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?![aA][mMwW][sS]:)[a-zA-Z0-9\\s_.:/=+\\\\\\\-@\\]*]+$",
            "minLength": 1,
            "maxLength": 127
```

```
            },
            "minItems": 1,
            "maxItems": 50,
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "ResourceArn",
            "AddOrUpdateTags",
            "RemoveTags"
          ]
        },
        "required": [
          "ResourceArn"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Resources",
      "Priority"
    ]
  },
  "required": [
    "Resources"
  ]
```

```
}
```

Schema for Change Type ct-1078jhyxq32dp

Classifications:

- Management | Directory Service | Computer object | Remove SPN

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove Service Principal Name",
  "description": "Remove the Service Principal Name (SPN) associated with a specified
 hostname or host alias in Microsoft Active Directory. For multi-account landing zone
 (MALZ), use this change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-RemoveADComputerSPN-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveADComputerSPN-Admin"
      ],
      "default": "AWSManagedServices-RemoveADComputerSPN-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Hostname": {
          "description": "The hostname of the computer tagged with the SPN.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
```

```
        },
        "ServiceType": {
          "description": "The type of service, such as MSSQLSvc, HTTP, TERMSRV, HOST,
 WSMAN, RestrictedKrbHost.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "MSSQLSvc",
              "HTTP",
              "TERMSRV",
              "HOST",
              "WSMAN",
              "RestrictedKrbHost"
            ],
            "default": "HOST"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "AliasName": {
          "description": "The alias associated with the host.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,15}$"
          }
        },
        "GroupManagedServiceAccountName": {
          "description": "The group Managed Service Account (gMSA) name used to run the
 specified ServiceType.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,15}$"
          }
        },
        "Port": {
          "description": "The port the service utilizes; for example, 1433.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\$?()([1-9]|[1-5]?[0-9]{2,4}|6[1-4][0-9]{3}|65[1-4][0-9]{2}|
655[1-2][0-9]|6553[1-5])$"
```

```
          }
        }
      },
      "metadata": {
        "ui:order": [
          "Hostname",
          "ServiceType",
          "AliasName",
          "GroupManagedServiceAccountName",
          "Port"
        ]
      },
      "additionalProperties": false,
      "required": [
        "Hostname",
        "ServiceType"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-111fhplhx9axe

Classifications:

- Management | Advanced stack components | Security group | Revoke egress rule

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
 "name": "Revoke Egress Rule",
 "description": "Revoke the egress rule for the specified security group (SG). You
must specify the configurations of the egress rule that you are revoking. Note that,
once revoked, the egress rule is permanently deleted.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-RevokeSecurityGroupEgressRule",
     "type": "string",
     "enum": [
       "AWSManagedServices-RevokeSecurityGroupEgressRule"
     ],
     "default": "AWSManagedServices-RevokeSecurityGroupEgressRule"
   },
   "Region": {
     "description": "The AWS Region in which the security group is located, in the
form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "SecurityGroupId": {
         "description": "The ID of the security group (SG) that you are updating, in
the form sg-0123456789abcdef.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "IpProtocol": {
         "description": "The IP protocol name, or IP protocol number, for the egress
rule. For example, for TCP, enter either TCP, or (IP protocol number) 6. If you enter
ICMP, you can specify any or all of the ICMP types and codes.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\+-\\\\(\\\\)\\w]{1,18}$"
         },
         "minItems": 1,
```

```
          "maxItems": 1
        },
        "FromPort": {
          "description": "Start of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ToPort": {
          "description": "End of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Destination": {
          "description": "An IP address, in the form 255.255.255.255, or an IP address
 range in CIDR notation, in the form 255.255.255.255/32, or the ID of another security
 group in the same region; or self to specify the same security group.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8}$|^sg-[0-9a-f]{17}$|^self$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SecurityGroupId",
```

```
          "IpProtocol",
          "FromPort",
          "ToPort",
          "Destination"
        ]
      },
      "required": [
        "SecurityGroupId",
        "IpProtocol",
        "FromPort",
        "ToPort",
        "Destination"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-111r1yayblnw4

Classifications:

- Deployment | Advanced stack components | Application Load Balancer | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Application Load Balancer",
  "description": "Create an AWS Application Load Balancer (ALB), with additional
 listeners.",
```

```
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "The ID of the VPC where you want the ALB, in the form
vpc-0123abcd or vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "minLength": 1,
           "maxLength": 255
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Key",
           "Value"
         ]
```

```
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-sd7uv500000000000",
      "type": "string",
      "enum": [
        "stm-sd7uv500000000000"
      ],
      "default": "stm-sd7uv500000000000"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 360
    },
    "LoadBalancer": {
      "type": "object",
      "properties": {
        "Name": {
          "type": "string",
          "description": "A friendly name for the load balancer. This name must be
unique per region per account, can have a maximum of 32 characters, must contain only
alphanumeric characters or hyphens, must not begin or end with a hyphen, and must
not begin with \"internal-\". If you don't specify a name a unique physical ID is
generated for the load balancer.",
          "pattern": "^(?!internal-)(?!-)([0-9a-zA-Z\\-]{0,32})[^\\-]$|^$"
        },
        "SecurityGroups": {
          "description": "A list of security groups to associate with the load
balancer.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "uniqueItems": true
        },
        "SubnetIds": {
          "description": "A list of subnet IDs that the Elastic Load Balancing creates
load balancer nodes in. You must specify subnets from at least two Availability Zones.
For an internet-facing load balancer provide a public subnet ID, for an internal load
balancer we recommend private subnet IDs.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 2,
          "uniqueItems": true
        },
        "Public": {
          "type": "string",
          "description": "True if the load balancer endpoint is public, false if it is
private.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "DeletionProtection": {
          "type": "string",
          "description": "True to enable deletion protection, false to not. Default is
false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "IdleTimeout": {
          "type": "string",
          "description": "How long the load balancer front-end connection (client to
load balancer) can be idle (not receiving data) before the connection is automatically
closed.",
          "pattern": "^([1-9][0-9]{0,2}|[1-3][0-9]{3}|4000)$",
```

```
          "default": "60"
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "Public",
          "SecurityGroups",
          "SubnetIds",
          "IdleTimeout",
          "DeletionProtection"
        ]
      },
      "required": [
        "SecurityGroups",
        "SubnetIds"
      ],
      "additionalProperties": false
    },
    "Listener1": {
      "type": "object",
      "properties": {
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
external incoming traffic.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$",
          "default": "80"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
connections (client to load balancer). The supported protocols are HTTP and HTTPS.",
          "enum": [
            "HTTP",
            "HTTPS"
          ],
          "default": "HTTP"
        },
        "SSLCertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the certificate to
associate with the listener, in the form arn:aws:acm:region:account-id:certificate/
```

```
certificate-id or arn:aws:iam::account-id:server-certificate/certificate-name. Leave
 blank if Protocol is not HTTPS.",
          "pattern": "^$|^(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})$|^(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$"
        },
        "SSLPolicy": {
          "type": "string",
          "description": "The security policy that defines the ciphers and protocols
 that the load balancer supports. Use only if Protocol = HTTPS. For details on default
 AWS security policies, see AWS documentation for ALBs.",
          "enum": [
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "SSLCertificateArn",
          "SSLPolicy"
        ]
      },
      "required": [
        "Port",
        "Protocol"
```

```
      ],
      "additionalProperties": false
    },
    "Listener2": {
      "type": "object",
      "properties": {
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$|^$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer). The supported protocols are HTTP and HTTPS.",
          "pattern": "^$|^(HTTP|HTTPS)$"
        },
        "SSLCertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the certificate to
 associate with the listener, in the form arn:aws:acm:region:account-id:certificate/
certificate-id or arn:aws:iam::account-id:server-certificate/certificate-name. Leave
 blank if Protocol is not HTTPS.",
          "pattern": "^$|^(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})$|^(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$"
        },
        "SSLPolicy": {
          "type": "string",
          "description": "The security policy that defines the ciphers and protocols
 that the load balancer supports. Use only if Protocol = HTTPS. See AWS documentation
 for ALBs for details on default AWS security policies.",
          "enum": [
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
```

```
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "SSLCertificateArn",
          "SSLPolicy"
        ]
      },
      "additionalProperties": false
    },
    "TargetGroup": {
      "type": "object",
      "properties": {
        "Name": {
          "type": "string",
          "description": "An optional friendly name for the target group. This name
must be unique per region per account, can have a maximum of 32 characters, must
contain only alphanumeric characters or hyphens, must not begin or end with a hyphen,
and must not begin with \"internal-\". If you don't specify a name a unique physical
ID is generated for the target group.",
          "pattern": "^(?!internal-)(?!-)([0-9a-zA-Z\\-]{0,32})[^\\-]$|^$",
          "default": ""
        },
        "HealthCheckInterval": {
          "type": "string",
          "description": "The approximate amount of time, in seconds, between health
checks of an individual target. The range is 5 to 300 seconds.",
          "pattern": "^([5-9]|[1-8][0-9]|9[0-9]|[12][0-9]{2}|300)$",
          "default": "10"
        },
        "HealthCheckPath": {
          "type": "string",
```

```
          "description": "The ping path destination where Elastic Load Balancing sends
 health check requests.",
          "default": "/",
          "pattern": "^(/?[a-z0-9\\-._~%!$&'()*+,;=@]+(/[a-z0-9\\-._~%!$&'()*
+,;=:@]+)*/?|/){1,1024}$"
        },
        "HealthCheckPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health
 checks on targets. The default is traffic-port, which is the port on which each target
 receives traffic from the load balancer.",
          "pattern": "^$|^([0-9]{1,5})$"
        },
        "HealthCheckProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS"
          ],
          "default": "HTTP"
        },
        "HealthCheckTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, to wait for a response to
 a health check. Must be less than the value for HealthCheckInterval. The supported
 values are 2 seconds to 60 seconds.",
          "pattern": "^(60|[1-5]{1}[0-9]{1}|[2-9]{1})$"
        },
        "HealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health probe successes required
 before moving the instance to the Healthy state.",
          "pattern": "^([2-9]{1}|10)$",
          "default": "2"
        },
        "UnhealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health probe failures required
 before moving the instance to the Unhealthy state.",
          "pattern": "^([2-9]{1}|10)$",
          "default": "10"
        },
```

```
        "ValidHTTPCode": {
          "type": "string",
          "description": "The HTTP codes that a healthy target application server must
 use in response to a health check. You can specify multiple values such as 200,202, or
 a range of values such as 200-499. Only applicable if HealthCheckTargetProtocol = HTTP
 or HTTPS.",
          "pattern": "^(([2-4]{1}[0-9]{2}($|-|,))+)$",
          "default": "200"
        },
        "TargetPort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$",
          "default": "80"
        },
        "TargetProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS"
          ],
          "default": "HTTP"
        },
        "DeregistrationDelayTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing
 to wait before changing the state of a deregistering target from draining to unused.
 Valid value ranges from 0 to 3600. The default value is 300 seconds.",
          "pattern": "^(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})$",
          "default": "300"
        },
        "SlowStartDuration": {
          "type": "string",
          "description": "The time period, in the range 30-900 seconds, during which
 the load balancer sends a newly registered target a linearly-increasing share of the
 target group traffic",
          "pattern": "^([3-9]{1}[0-9]{1}|[1-8]{1}[0-9]{2}|900|0)$|^$"
        },
        "CookieExpirationPeriod": {
          "type": "string",
```

```
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "pattern": "^([1-9]{1}[0-9]{0,4}|[1-5]{1}[0-9]{5}|60[0-3]{1}[0-9]{3}|604[0-7]
{1}[0-9]{2}|604800)$|^$"
        },
        "TargetType": {
          "type": "string",
          "description": "The type of target that you must specify when registering
 targets with this target group.",
          "enum": [
            "instance",
            "ip"
          ],
          "default": "instance"
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "TargetType",
          "TargetPort",
          "TargetProtocol",
          "HealthCheckInterval",
          "HealthCheckPath",
          "HealthCheckPort",
          "HealthCheckProtocol",
          "HealthCheckTimeout",
          "HealthyThreshold",
          "UnhealthyThreshold",
          "ValidHTTPCode",
          "DeregistrationDelayTimeout",
          "SlowStartDuration",
          "CookieExpirationPeriod"
        ]
      },
      "additionalProperties": false
    },
    "HealthyHostsAlarm": {
      "type": "object",
      "properties": {
        "EvaluationPeriods": {
          "type": "string",
```

```
          "description": "The number of the most recent periods to evaluate when
 determining alarm state. The valid number of period intervals is any integer greater
 than 0 and the default value is 5.",
          "pattern": "^([1-9]|[1-9][0-9]{1,})$",
          "default": "5"
        },
        "Period": {
          "type": "string",
          "description": "The period, in seconds, over which to evaluate the
 HealthyHostCount metric. Valid values are any multiple of 60 (including 60). The
 default value is 60 seconds.",
          "pattern": "^(6[0]+|12[0]+|18[0]+|24[0]+|30[0]+|36[0]+|42[0]+|48[0]+|
54[0]+)$",
          "default": "60"
        },
        "Threshold": {
          "type": "string",
          "description": "The minimum number of healthy instances associated to the
 load balancer within an evaluation period for the alarm to trigger. 0 means at least 1
 healthy instance required for not alarming.",
          "pattern": "^([0-9](\\.0)|[1-9][0-9]{1,}(\\.0))$",
          "default": "0.0"
        }
      },
      "metadata": {
        "ui:order": [
          "EvaluationPeriods",
          "Period",
          "Threshold"
        ]
      },
      "additionalProperties": false
    },
    "HTTPCodeELB5XXCountAlarm": {
      "type": "object",
      "properties": {
        "EvaluationPeriods": {
          "type": "string",
          "description": "The number of the most recent periods to evaluate when
 determining alarm state. The valid number of period intervals is any integer greater
 than 0 and the default value is 3.",
          "pattern": "^([1-9]|[1-9][0-9]{1,})$",
          "default": "3"
        },
```

```
        "Period": {
          "type": "string",
          "description": "The period, in seconds, over which to evaluate the
 HTTPCode_ELB_5XX_Count metric. Valid values are any multiple of 60 (including 60). The
 default value is 300 seconds.",
          "pattern": "^(6[0]+|12[0]+|18[0]+|24[0]+|30[0]+|36[0]+|42[0]+|48[0]+|
54[0]+)$",
          "default": "300"
        },
        "Threshold": {
          "type": "string",
          "description": "The number of HTTP 5XX server error codes that originate from
 the load balancer that must be exceedeed within an evaluation period for the alarm to
 trigger.",
          "pattern": "^([0-9](\\.0)|[1-9][0-9]{1,}(\\.0))$",
          "default": "0.0"
        }
      },
      "metadata": {
        "ui:order": [
          "EvaluationPeriods",
          "Period",
          "Threshold"
        ]
      },
      "additionalProperties": false
    },
    "TargetConnectionErrorsAlarm": {
      "type": "object",
      "properties": {
        "EvaluationPeriods": {
          "type": "string",
          "description": "The number of the most recent periods to evaluate when
 determining alarm state. The valid number of period intervals is any integer greater
 than 0 and the default value is 3.",
          "pattern": "^([1-9]|[1-9][0-9]{1,})$",
          "default": "3"
        },
        "Period": {
          "type": "string",
          "description": "The period, in seconds, over which to evaluate the
 TargetConnectionErrorCount metric. Valid values are any multiple of 60 (including 60).
 The default value is 300 seconds.",
```

```
          "pattern": "^(6[0]+|12[0]+|18[0]+|24[0]+|30[0]+|36[0]+|42[0]+|48[0]+|
54[0]+)$",
          "default": "300"
        },
        "Threshold": {
          "type": "string",
          "description": "The number of unsuccessful connections between the load
 balancer and the Target Group that must be exceedeed within an evaluation period for
 the alarm to trigger.",
          "pattern": "^([0-9](\\.0)|[1-9][0-9]{1,}(\\.0))$",
          "default": "0.0"
        }
      },
      "metadata": {
        "ui:order": [
          "EvaluationPeriods",
          "Period",
          "Threshold"
        ]
      },
      "additionalProperties": false
    },
    "RejectedConnectionCountAlarm": {
      "type": "object",
      "properties": {
        "EvaluationPeriods": {
          "type": "string",
          "description": "The number of the most recent periods to evaluate when
 determining alarm state. The valid number of period intervals is any integer greater
 than 0 and the default value is 5.",
          "pattern": "^([1-9]|[1-9][0-9]{1,})$",
          "default": "5"
        },
        "Period": {
          "type": "string",
          "description": "The period, in seconds, over which to evaluate the
 RejectedConnectionCount metric. Valid values are any multiple of 60 (including 60).
 The default value is 60 seconds.",
          "pattern": "^(6[0]+|12[0]+|18[0]+|24[0]+|30[0]+|36[0]+|42[0]+|48[0]+|
54[0]+)$",
          "default": "60"
        },
        "Threshold": {
          "type": "string",
```

```
          "description": "The number of rejected connections (due to reaching service
limits) that originate from the load balancer that must be exceedeed within an
evaluation period for the alarm to trigger.",
          "pattern": "^([0-9](\\.0)|[1-9][0-9]{1,}(\\.0))$",
          "default": "0.0"
        }
      },
      "metadata": {
        "ui:order": [
          "EvaluationPeriods",
          "Period",
          "Threshold"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags",
      "LoadBalancer",
      "Listener1",
      "Listener2",
      "TargetGroup",
      "HealthyHostsAlarm",
      "HTTPCodeELB5XXCountAlarm",
      "TargetConnectionErrorsAlarm",
      "RejectedConnectionCountAlarm"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "StackTemplateId",
    "LoadBalancer",
    "Listener1"
  ],
```

```
  "additionalProperties": false
}
```

Schema for Change Type ct-117rmp64d5mvb

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create EC2 instance profile

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EC2 Instance Profile",
  "description": "Create an IAM instance profile to use with EC2 instances. Each ARN
 specified in the parameters creates a part of the IAM policy. Use the Preview option
 to see what the completed, generated, policy looks like before it is created and
 implemented.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleCreateIAMRole-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleCreateIAMRole-Admin"
      ],
      "default": "AWSManagedServices-HandleCreateIAMRole-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
```

```
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ServicePrincipal": {
          "description": "Must be ec2.amazonaws.com. This establishes the trust
 relationship with the EC2 service for this role.",
          "type": "string",
          "enum": [
            "ec2.amazonaws.com"
          ],
          "default": "ec2.amazonaws.com"
        },
        "RoleName": {
          "description": "A name for the IAM role. The name can be up to 64 characters
 in length and is limited to use characters a-z, A-Z, 0-9, and _+=,.@-.",
          "type": "string",
          "pattern": "(?![aA][mMwW][sS]|customer-mc|managementhost|ms-)[a-zA-Z0-9_
+=,.@-]{1,64}$"
        },
        "RolePath": {
          "description": "A path for the IAM role, a string of characters consisting of
 either a forward slash (/) by itself or a string that must begin and end with forward
 slash (/).",
          "type": "string",
          "default": "/",
          "pattern": "^\\/{1}([^\\/]*\\/)?$"
```

```
        },
        "Preview": {
          "description": "Yes to preview the IAM role policy created with the specified
 parameter values, without creating the role; No to not preview it but to create and
 implement the role. The preview is provided as a JSON in the execution output. In
 order to implement the policy after preview, create a copy of the RFC and set the
 Preview parameter to No, then submit.",
          "type": "string",
          "default": "No",
          "enum": [
            "Yes",
            "No"
          ]
        },
        "S3ReadAccess": {
          "description": "A list of Amazon resource names (ARNs) of S3 buckets. Scopes
 down the policy for S3 read access to the given buckets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(^arn:(aws|aws-us-gov):s3:::.+$)|(^$)"
          },
          "maxItems": 50
        },
        "S3WriteAccess": {
          "description": "A list of S3 bucket ARNs. Scopes down the policy for S3 write
 access to the given buckets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(^arn:(aws|aws-us-gov):s3:::.+$)|^[*]$|(^$)"
          },
          "maxItems": 50
        },
        "KMSReadAccess": {
          "description": "A list of KMS key ARNs. Scopes down the policy for KMS read
 access to the given KMS keys only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:(key|alias)/.
+)$|^$"
          },
          "maxItems": 50
```

```
        },
        "KMSCryptographicOperationAccess": {
          "description": "A list of KMS key ARNs. Scopes down the policy for
 cryptographic operation access to the given ARNs only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/[a-f0-9]{8}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
          },
          "maxItems": 50
        },
        "SSMReadAccess": {
          "description": "A list of SSM parameter ARNs. Scopes down the policy for SSM
 read access to the given parameters only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):ssm:[a-z0-9-]+:[0-9]{12}:parameter/.+)$|
^$"
          },
          "maxItems": 50
        },
        "SSMWriteAccess": {
          "description": "A list of SSM parameter ARNs. Scopes down the policy for SSM
 write access to given parameters only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):ssm:[a-z0-9-]+:[0-9]{12}:parameter/.+)$|
^$"
          },
          "maxItems": 50
        },
        "CloudWatchLogsReadAccess": {
          "description": "A list of CloudWatch resource ARNs. Scopes down the policy
 for read access to given CloudWatch Logs resource only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):logs:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^
$"
          },
          "maxItems": 50
```

```
        },
        "CloudWatchLogsWriteAccess": {
          "description": "A list of CloudWatch resource ARNs. Scopes down the policy
 for write access to given CloudWatch Logs resource only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):logs:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchAlarmReadAccess": {
          "description": "A list of CloudWatch alarm ARNs. Scopes down the policy for
 read access to given CloudWatch alarms only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):cloudwatch:[a-z0-9-]+:[0-9]{12}:alarm:.
+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchAlarmWriteAccess": {
          "description": "A list of CloudWatch alarm ARNs. Scopes down the policy for
 write access to given CloudWatch alarms only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):cloudwatch:[a-z0-9-]+:[0-9]{12}:alarm:.
+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchMetricsReadAccess": {
          "description": "For read access to metrics, use an asterisk ( * ). Scopes
 down the policy for read access to all CloudWatch metrics.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[*]$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchMetricsWriteAccess": {
```

```
          "description": "A list of CloudWatch metric namespaces. Scopes down the
 policy for write access to given CoudWatch metric namespaces only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(.*?)|^$"
          },
          "maxItems": 50
        },
        "SecretsManagerReadAccess": {
          "description": "A list of Secrets Manager secret ARNs. Scopes down the policy
 for read access to given secrets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):secretsmanager:[a-z0-9-]+:[0-9]
{12}:secret:.+)$|^$"
          },
          "maxItems": 50
        },
        "SNSReadAccess": {
          "description": "A list of SNS resource ARNs. Scopes down the policy for SNS
 read access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sns:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^$"
          },
          "maxItems": 50
        },
        "SNSWriteAccess": {
          "description": "A list of SNS resource ARNs. Scopes down the policy for SNS
 write access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sns:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
        },
        "SQSReadAccess": {
          "description": "A list of SQS resource ARNs. Scopes down the policy for SQS
 read access to given resources only.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sqs:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^$"
          },
          "maxItems": 50
        },
        "SQSWriteAccess": {
          "description": "A list of SQS resource ARNs. Scopes down the policy for SQS
 write access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sqs:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
        },
        "DynamoDBResourceReadAccess": {
          "description": "A list of DynamoDB resource ARNs. Scopes down the policy for
 DynamoDB read access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):dynamodb:[a-z0-9-]+:[0-9]{12}:.+)$|
^[*]$|^$"
          },
          "maxItems": 50
        },
        "DynamoDBDataReadWriteAccess": {
          "description": "A list of DynamoDB table ARNs. Scopes down the policy for
 DynamoDB data read and write access to given tables only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):dynamodb:[a-z0-9-]+:[0-9]{12}:table/.
+)$|^$"
          },
          "maxItems": 50
        },
        "STSAssumeRole": {
          "description": "A list of IAM role ARNs. Scopes down the policy for STS
 assume role to given IAM roles only.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^(arn:(aws|aws-us-gov):iam::[0-9]{12}:role/.+)$|^$"
          },
          "maxItems": 50
        },
        "AdditionalPolicy": {
          "description": "An additional policy document as a JSON that is less
permissive than the AMS baseline policy. For details on AMS baseline policy see AMS
documentation.",
          "type": "string",
          "pattern": "^[\\s\\S]*$",
          "maxLength": 10240
        }
      },
      "metadata": {
        "ui:order": [
          "ServicePrincipal",
          "RoleName",
          "RolePath",
          "Preview",
          "S3ReadAccess",
          "S3WriteAccess",
          "KMSReadAccess",
          "KMSCryptographicOperationAccess",
          "SSMReadAccess",
          "SSMWriteAccess",
          "CloudWatchLogsReadAccess",
          "CloudWatchLogsWriteAccess",
          "CloudWatchAlarmReadAccess",
          "CloudWatchAlarmWriteAccess",
          "CloudWatchMetricsReadAccess",
          "CloudWatchMetricsWriteAccess",
          "SecretsManagerReadAccess",
          "SNSReadAccess",
          "SNSWriteAccess",
          "SQSReadAccess",
          "SQSWriteAccess",
          "DynamoDBResourceReadAccess",
          "DynamoDBDataReadWriteAccess",
          "STSAssumeRole",
          "AdditionalPolicy"
        ]
      },
      "required": [
        "ServicePrincipal",
```

```
        "RoleName",
        "Preview"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-128svy9nn2yj8

Classifications:

- Management | Advanced stack components | S3 storage | Update encryption

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change S3 Bucket Encryption Setting",
  "description": "Enable or update S3 bucket encryption setting through direct API
 calls. The S3 bucket can be standalone or belong to a CloudFormation stack; in the
 latter case, the change might cause stack drift. To avoid causing stack drift, please
 use ct-1gi93jhvj28eg instead, or ct-361tlo1k7339x if the S3 bucket was provisioned via
 CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateBucketEncryption.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateBucketEncryption"
```

```
      ],
      "default": "AWSManagedServices-UpdateBucketEncryption"
    },
    "Region": {
      "description": "The AWS Region in which the resource is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BucketName": {
          "description": "The name of the bucket for which to update the encryption
 setting.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(mc|ams|awsms)-)[a-z0-9][-.a-z0-9]{1,61}[a-z0-9]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ServerSideEncryption": {
          "description": "Default encryption for an S3 bucket using server-side
 encryption with either Amazon S3-managed keys (SSE-S3) or AWS KMS-managed keys (SSE-
KMS).",
          "type": "string",
          "enum": [
            "S3ManagedKeys",
            "KmsManagedKeys"
          ]
        },
        "KMSKeyId": {
          "description": "The AWS KMS master key ID used for the ServerSideEncryption
 KMS encryption. Applicable only if ServerSideEncryption = KmsManagedKeys.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/)?
[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^(arn:(aws|aws-cn|aws-
us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/)?mrk-[a-f0-9]{33}$|^(arn:(aws|aws-cn|aws-us-
gov):kms:[a-z0-9-]+:[0-9]{12}:)?alias/.{1,}$|^$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "BucketName",
          "ServerSideEncryption",
          "KMSKeyId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BucketName",
        "ServerSideEncryption"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-12amsdz909cfh

Classifications:

- Deployment | Advanced stack components | Load balancer (ELB) stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create load balancer (ELB) stack",
```

```
 "description": "Use to create an Amazon ELB Classic Load Balancer. Use alternate
change types to create an Application Load Balancer (ct-111r1yayblnw4) or Network Load
Balancer (ct-2qldv4h9osmau).",
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the vpc to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "StackTemplateId": {
     "description": "Must be stm-sdhopv30000000000.",
     "type": "string",
     "enum": [
       "stm-sdhopv30000000000"
     ]
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
```

```
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "ELBBackendInstances": {
          "default": [
            ""
          ],
          "description": "One or more EC2 instance IDs to associate with the load
balancer, in the form of i-0123abcd or i-01234567890abcdef for a single instance,
or i-0123abcd,i-12345abcd or i-01234567890abcdef,i-2345678901abcdefg for multiple
instances. Leave blank to not associate individual EC2 instances with the load
balancer. A load balancer can be associated with an autoscaling group by specifying
the load balancer name in the ASGLoadBalancerNames property during creation or update
of the autoscaling group.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8}$|^i-[a-z0-9]{17}$|^$"
          },
          "uniqueItems": true
        },
        "ELBCrossZone": {
          "description": "With cross-zone load balancing, your load balancer nodes
route traffic to the back-end instances across all Availability Zones. True to enable,
false to disable. The default is true.",
          "type": "boolean",
          "default": true
        },
        "ELBCookieExpirationPeriod": {
          "default": "",
          "description": "The time period, in seconds, after which the cookie is
considered stale. If this parameter isn't specified, the sticky session lasts for the
duration of the browser session.",
          "type": "string",
          "pattern": "^[0-9]+$|^$"
        },
        "ELBCookieExpirationPeriod2": {
          "default": "",
          "description": "The time period, in seconds, after which the cookie is
considered stale. If this parameter isn't specified, the sticky session lasts for the
duration of the browser session.",
          "type": "string",
          "pattern": "^[0-9]+$|^$"
        },
        "ELBCookieStickinessPolicyName": {
          "default": "",
          "description": "A name for the cookie stickiness policy. The name must be
unique within the set of policies for this load balancer. Leave blank to skip creation
of a policy.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "ELBCookieStickinessPolicyName2": {
          "default": "",
          "description": "A name for the second cookie stickiness policy. The name
must be unique within the set of policies for this load balancer. Leave blank to skip
creation of a second policy.",
          "type": "string",
```

```
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
        },
        "ELBSubnetIds": {
          "description": "One or more subnet IDs for the load balancer, in the form
 subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "ELBHealthCheckHealthyThreshold": {
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "type": "number",
          "minimum": 2,
          "maximum": 10,
          "default": 10
        },
        "ELBHealthCheckInterval": {
          "description": "The approximate interval, in seconds, between health
 checks.",
          "type": "number",
          "minimum": 5,
          "maximum": 300,
          "default": 30
        },
        "ELBHealthCheckTarget": {
          "description": "The protocol, port, and path of the instance to check. For
 example, HTTP:80/weather/us/wa/seattle. The protocol can be TCP, HTTP, HTTPS, or SSL.
 The range of valid ports is 1 through 65535.",
          "type": "string",
          "pattern": "^(HTTP|HTTPS):[0-9]{1,5}[/][a-zA-Z0-9/_.-]*$|^(SSL|TCP):[0-9]
{1,5}$"
        },
        "ELBHealthCheckTimeout": {
          "description": "The amount of time, in seconds, to wait for a response to a
 health check. Must be less than the value for ELBHealthCheckInterval.",
          "type": "number",
          "minimum": 2,
          "maximum": 60,
          "default": 5
```

```
        },
        "ELBHealthCheckUnhealthyThreshold": {
          "description": "The number of consecutive health check failures required to
 declare an EC2 instance unhealthy.",
          "type": "number",
          "minimum": 2,
          "maximum": 10,
          "default": 2
        },
        "ELBIdleTimeout": {
          "description": "The time, in seconds, that a connection to the load balancer
 can remain idle, which means no data is sent over the connection. After the specified
 time, the load balancer closes the connection.",
          "type": "number",
          "minimum": 1,
          "maximum": 3600,
          "default": 60
        },
        "ELBInstancePort": {
          "default": "80",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBInstancePort2": {
          "default": "80",
          "description": "The TCP port the optional second listener uses to send
 traffic to the target instance.",
          "type": "string",
          "pattern": "^[0-9]{1,5}$"
        },
        "ELBInstanceProtocol": {
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "ELBInstanceProtocol2": {
```

```
        "description": "The protocol the second listener uses for routing traffic to
back-end connections (load balancer to backend instance).",
        "type": "string",
        "enum": [
          "HTTP",
          "HTTPS",
          "SSL",
          "TCP"
        ]
      },
      "ELBLoadBalancerName": {
        "description": "A friendly name for the load balancer.",
        "type": "string",
        "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,31}$|^$"
      },
      "ELBLoadBalancerPort": {
        "default": "80",
        "description": "The port number for the load balancer to use when routing
external incoming traffic.",
        "type": "string",
        "pattern": "^[0-9]{1,5}$"
      },
      "ELBLoadBalancerPort2": {
        "default": "81",
        "description": "The port number for the load balancer to use when routing
external incoming traffic on the second listener.",
        "type": "string",
        "pattern": "^[0-9]{1,5}$"
      },
      "ELBLoadBalancerProtocol": {
        "default": "HTTP",
        "description": "The transport protocol to use for routing front-end
connections (client to load balancer).",
        "type": "string",
        "enum": [
          "HTTP",
          "HTTPS",
          "SSL",
          "TCP"
        ]
      },
      "ELBLoadBalancerProtocol2": {
```

```
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) on the second listener. Leave blank to skip
 creation of an additional listener.",
          "type": "string",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "ELBScheme": {
          "description": "True if the load balancer endpoint is public, false if it
 is not. Default is false. Set to true if you choose a public subnet for the load
 balancer.",
          "type": "boolean",
          "default": false
        },
        "ELBSSLCertificateId": {
          "default": "",
          "description": "The Amazon Resource Name (ARN)
 of the SSL certificate to use, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012. This must be
 specified if the HTTPS or SSL protocol is specified for ELBLoadBalancerProtocol.",
          "type": "string",
          "pattern": "^$|(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})|(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$|^([a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]
{12})$"
        },
        "ELBSSLCertificateId2": {
          "default": "",
          "description": "The Amazon Resource Name (ARN) of the SSL certificate
 to use for the optional second listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012. Leave blank if a
 second listener is not being created or if the second listener does not use the HTTPS
 or SSL for ELBLoadBalancerProtocol2.",
          "type": "string",
          "pattern": "^$|(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})|(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$|^([a-z0-9]{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]
{12})$"
        }
      },
```

```
      "metadata": {
        "ui:order": [
          "ELBSubnetIds",
          "ELBLoadBalancerName",
          "ELBScheme",
          "ELBBackendInstances",
          "ELBIdleTimeout",
          "ELBCrossZone",
          "ELBHealthCheckTarget",
          "ELBHealthCheckInterval",
          "ELBHealthCheckTimeout",
          "ELBHealthCheckHealthyThreshold",
          "ELBHealthCheckUnhealthyThreshold",
          "ELBCookieStickinessPolicyName",
          "ELBCookieExpirationPeriod",
          "ELBInstancePort",
          "ELBInstanceProtocol",
          "ELBLoadBalancerPort",
          "ELBLoadBalancerProtocol",
          "ELBSSLCertificateId",
          "ELBCookieExpirationPeriod2",
          "ELBCookieStickinessPolicyName2",
          "ELBInstancePort2",
          "ELBInstanceProtocol2",
          "ELBLoadBalancerPort2",
          "ELBLoadBalancerProtocol2",
          "ELBSSLCertificateId2"
        ]
      },
      "required": [
        "ELBSubnetIds",
        "ELBLoadBalancerPort",
        "ELBLoadBalancerProtocol",
        "ELBInstancePort"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
```

```
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-12lyw7otiyr6f

Classifications:

- Management | Advanced stack components | Security group | Associate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Associate Security Group",
  "description": "Associate security groups with an AWS resource.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AttachSecurityGroupsV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AttachSecurityGroupsV2"
      ],
      "default": "AWSManagedServices-AttachSecurityGroupsV2"
    },
    "Region": {
      "description": "The AWS Region in which the security groups are located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
```

```
   "Parameters": {
     "type": "object",
     "properties": {
       "ResourceType": {
         "description": "The type of resource to associate the
security group or groups to. Supported resource types are EC2Instance,
ElasticNetworkInterface, AutoScalingGroup, AutoScalingGroupCurrentInstancesOnly,
ElasticLoadBalancer, ApplicationLoadBalancer, RDSDBInstance, RDSDBCluster,
ElasticacheCluster, RedshiftCluster, ElasticFileSystem. Important Note: For
AutoScalingGroupCurrentInstancesOnly, security groups are only attached to individual
instances currently part of the ASG. LaunchTemplate or LaunchConfiguration are not
updated. Please make sure to update LaunchTemplate / LaunchConfiguration before
updating security groups to AutoScalingGroup Instances.",
         "type": "string",
         "enum": [
           "EC2Instance",
           "ElasticNetworkInterface",
           "AutoScalingGroup",
           "AutoScalingGroupCurrentInstancesOnly",
           "ElasticLoadBalancer",
           "ApplicationLoadBalancer",
           "RDSDBInstance",
           "RDSDBCluster",
           "ElasticacheCluster",
           "RedshiftCluster",
           "ElasticFileSystem"
         ]
       },
       "ResourceId": {
         "description": "The resource identifier to associate the security
groups to, per specified ResourceType. For EC2Iinstance use the instance ID,
for ElasticNetworkInterface use the network interface ID, for AutoScalingGroup
and AutoScalingGroupCurrentInstancesOnly use the Auto Scaling group name, for
ElasticLoadBalancer use the load balancer name; for ApplicationLoadBalancer use the
load balancer ARN or the load balancer name; for RDSDBInstance use the DB instance ID;
for RDSDBCluster use the DB cluster ID, for ElasticacheCluster use the cache cluster
ID, for RedshiftCluster use the cluster ID, for ElasticFileSystem use file system
Id.",
         "type": "string",
         "pattern": "^.+$"
       },
       "SecurityGroupIds": {
         "description": "A list of security group IDs to associate to the specified
ResourceId.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-([0-9a-f]{8}|[0-9a-f]{17})$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "OverwriteSecurityGroups": {
          "description": "True to overwrite the existing security groups of the
resource with the specified SecurityGroupIds, false to not overwrite the existing
list. Default is false and existing security groups are retained. IMPORTANT: If true,
any access allowed by existing security groups is removed and only the new security
groups are in effect.",
          "type": "string",
          "default": "false",
          "enum": [
            "true",
            "false"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "ResourceType",
          "ResourceId",
          "SecurityGroupIds",
          "OverwriteSecurityGroups"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResourceType",
        "ResourceId",
        "SecurityGroupIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
```

```
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-12w49boaiwtzp

Classifications:

- Management | Advanced stack components | RDS database stack | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update RDS database stack",
  "description": "Modify the properties of an Amazon Relational Database Service (RDS)
 DB instance created using ct-2z60dyvto9g6c, version 3.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC that contains the RDS DB instance, in the form
 vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the RDS DB instance that you are updating, in the
 form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the RDS DB instance.",
      "type": "object",
      "properties": {
        "RDSAllocatedStorage": {
```

```
        "description": "The size of the database in gigabytes (GB). The acceptable
limits for this value relate to the engine and storage type that you specify. For
details, see AWS documentation on DB instance storage.",
        "type": "number",
        "minimum": 20,
        "maximum": 32768
      },
      "RDSAllowMajorVersionUpgrade": {
        "description": "True to allow updates to the DB instance's major version.",
        "type": "boolean"
      },
      "RDSAutoMinorVersionUpgrade": {
        "description": "True to apply minor engine upgrades automatically to the DB
instance during the maintenance window.",
        "type": "boolean"
      },
      "RDSBackupRetentionPeriod": {
        "description": "The number of days to retain automatic DB snapshots. Setting
this to a positive number enables backups. Setting this to 0 disables automated
backups.",
        "type": "number",
        "minimum": 0,
        "maximum": 35
      },
      "RDSDBParameterGroupName": {
        "description": "The name of an existing DB parameter group. If any of the
data members of the referenced parameter group are changed during an update, the DB
instance might need to be restarted, which causes some interruption. If the parameter
group contains static parameters, whether they were changed or not, an update triggers
a reboot.",
        "type": "string"
      },
      "RDSDeletionProtection": {
        "description": "True to disable DB instance deletion.",
        "type": "boolean"
      },
      "RDSDomain": {
        "description": "The Active Directory directory ID to create the instance in.
This is applicable only for Microsoft SQL Server DB engines only.",
        "type": "string",
        "pattern": "^$|^d-[0-9a-f]{10}$"
      },
      "RDSDomainIAMRoleName": {
```

```
        "description": "The name of an IAM role that Amazon RDS uses when calling
the AWS Directory Service APIs. This is applicable only for Microsoft SQL Server DB
engines only.",
        "type": "string",
        "pattern": "^$|^customer[\\w-]+$"
      },
      "RDSEngineVersion": {
        "description": "The version number of the database engine to use. Changing
this parameter results in DB instance restart.",
        "type": "string"
      },
      "RDSInstanceType": {
        "description": "The compute and memory capacity for the DB instance.",
        "type": "string"
      },
      "RDSIOPS": {
        "description": "The provisioned IOPS for RDS storage. Must be a multiple
between 3 and 10 of the storage amount for the DB instance. Must also be an integer
multiple of 1000. For example, if the size of your DB instance is 500 GB, then your
IOPS value can be 2000, 3000, 4000, or 5000.",
        "type": "number"
      },
      "RDSMasterUserPassword": {
        "description": "The password that you will use with the configured user name
to log in to your DB instance. Must contain from 8 to 30 printable ASCII characters
(excluding backslash, double quotes, and at sign).",
        "type": "string",
        "pattern": "^[!#-.0-?A-~]{8,30}$",
        "metadata": {
          "ams:sensitive": true
        }
      },
      "RDSMultiAZ": {
        "description": "True to have a standby replica of your DB instance created
in another Availability Zone for failover support, false to not have a standby
replica.",
        "type": "boolean"
      },
      "RDSPerformanceInsights": {
        "type": "string",
        "description": "True to enable Performance Insights for the DB instance,
false to not. Amazon RDS Performance Insights is a database performance tuning and
monitoring feature that helps you assess the load on your database.",
        "enum": [
```

```
            "true",
            "false"
          ]
        },
        "RDSPerformanceInsightsKMSKey": {
          "type": "string",
          "description": "The Amazon resource name (ARN) of the KMS master key to
 use to encrypt Performance Insights data. Specify default to use the default RDS KMS
 Key.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "RDSPerformanceInsightsRetentionPeriod": {
          "type": "string",
          "description": "The amount of time, in days, to retain Performance Insights
 data. Valid values are 7 or 731 (2 years).",
          "enum": [
            "7",
            "731"
          ]
        },
        "RDSOptionGroupName": {
          "description": "The option group that this DB instance is associated with.",
          "type": "string"
        },
        "RDSPreferredBackupWindow": {
          "description": "The daily time range during which automated backups are
 created, if RDSBackupRetentionPeriod is set to a positive number. Must be in the
 format hh:mm-hh:mm (24-hour format), in Universal Coordinated Time (UTC). Must not
 conflict with the RDSPreferredMaintenanceWindow setting, and must be at least 30
 minutes.",
          "type": "string",
          "pattern": "^$|^[0-9]{2}:[0-9]{2}-[0-9]{2}:[0-9]{2}$"
        },
        "RDSPreferredMaintenanceWindow": {
          "description": "The weekly time range during which system maintenance can
 occur, in UTC. Must be in the format ddd:hh:mm-ddd:hh:mm (24-hour format).",
          "type": "string",
          "pattern": "^$|^[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$"
        },
        "RDSStorageType": {
          "description": "Storage type for the RDS DB instance. If you specify io1, you
 must also include a value for the RDSIOPS parameter.",
          "type": "string",
```

```
          "enum": [
            "standard",
            "gp2",
            "io1",
            "gp3"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "RDSEngineVersion",
          "RDSInstanceType",
          "RDSStorageType",
          "RDSAllocatedStorage",
          "RDSIOPS",
          "RDSMasterUserPassword",
          "RDSMultiAZ",
          "RDSPerformanceInsights",
          "RDSPerformanceInsightsKMSKey",
          "RDSPerformanceInsightsRetentionPeriod",
          "RDSDomain",
          "RDSDomainIAMRoleName",
          "RDSDBParameterGroupName",
          "RDSOptionGroupName",
          "RDSBackupRetentionPeriod",
          "RDSPreferredBackupWindow",
          "RDSAutoMinorVersionUpgrade",
          "RDSAllowMajorVersionUpgrade",
          "RDSPreferredMaintenanceWindow",
          "RDSDeletionProtection"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
```

```
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-13lk0noacn6ua

Classifications:

- Management | Advanced stack components | Security group | Disassociate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disassociate Security Group",
  "description": "Disassociate a security group from up to 50 AWS resources and
 optionally delete the security group. This change type does not require a review and
 can be used instead of the manual, review required, change type (ct-3cp96z7r065e4).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DisassociateSecurityGroupV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DisassociateSecurityGroupV2"
      ],
      "default": "AWSManagedServices-DisassociateSecurityGroupV2"
    },
    "Region": {
      "description": "The AWS Region in which the security group is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityGroupId": {
          "description": "A security group ID to be disassociated from AWS resources.
 Provide at least one of EC2 instance IDs, Elastic network interface IDs, Auto scaling
 group names, Elastic load balancer names, Application load balancer names, RDS DB
 instance identifiers, RDS DB cluster identifiers, Elasticache cluster identifiers,
```

```
Redshift cluster identifiers, Elastic Filesystem identifiers to disassociate the
security group from.",
          "type": "string",
          "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
        },
        "EC2InstanceIds": {
          "description": "A list of up to 50 EC2 instance IDs to disassociate the
security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8}$|^i-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "ElasticNetworkInterfaceIds": {
          "description": "A list of up to 50 elastic network interface IDs to
disassociate the security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^eni-[a-z0-9]{8}$|^eni-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "AutoScalingGroupNames": {
          "description": "A list of up to 50 Auto scaling group names to disassociate
the security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-)).{1,255}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "ElasticLoadBalancerNames": {
          "description": "A list of up to 50 elastic load balancer names to
disassociate the security group from.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-zA-Z0-9][a-zA-Z0-9-]{1,30}[a-zA-Z0-9]$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "ApplicationLoadBalancerNames": {
          "description": "A list of up to 50 application load balancer names to
disassociate the security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-zA-Z0-9][a-zA-Z0-9-]{1,30}[a-zA-Z0-9]$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "RDSDBInstanceIdentifiers": {
          "description": "A list of up to 50 RDS DB instance identifiers to
disassociate the security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-zA-Z][a-zA-Z0-9-]{1,62}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "RDSDBClusterIdentifiers": {
          "description": "A list of up to 50 RDS DB cluster identifiers to disassociate
the security group from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-zA-Z][a-zA-Z0-9-]{1,62}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
```

```
      },
      "ElasticacheClusterIdentifiers": {
        "description": "A list of up to 50 Elasticache cluster identifiers to
disassociate the security group from.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$"
        },
        "minItems": 1,
        "maxItems": 50,
        "uniqueItems": true
      },
      "RedshiftClusterIdentifiers": {
        "description": "A list of up to 50 Redshift cluster identifiers to
disassociate the security group from.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$"
        },
        "minItems": 1,
        "maxItems": 50,
        "uniqueItems": true
      },
      "ElasticFileSystemIds": {
        "description": "A list of up to 50 Elastic file system identifiers to
disassociate the SecurityGroupId from.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$"
        },
        "minItems": 1,
        "maxItems": 50,
        "uniqueItems": true
      }
    },
    "metadata": {
      "ui:order": [
        "SecurityGroupId",
        "EC2InstanceIds",
        "ElasticNetworkInterfaceIds",
        "AutoScalingGroupNames",
```

```
          "ElasticLoadBalancerNames",
          "ApplicationLoadBalancerNames",
          "RDSDBInstanceIdentifiers",
          "RDSDBClusterIdentifiers",
          "ElasticacheClusterIdentifiers",
          "RedshiftClusterIdentifiers",
          "ElasticFileSystemIds"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SecurityGroupId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-13swbwdxg106z

Classifications:

- Management | Advanced stack components | RDS database stack | Update instance type

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Instance Type",
  "description": "Change the DB instance type through direct API calls. The RDS
 instance can be standalone or belong to a CloudFormation stack; in the latter
 case, the change might cause stack drift. To avoid causing stack drift, please use
```

```
 ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS instance was provisioned via
 CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateRDSInstanceType.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateRDSInstanceType"
      ],
      "default": "AWSManagedServices-UpdateRDSInstanceType"
    },
    "Region": {
      "description": "The AWS Region in which the resource is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBInstanceIdentifier": {
          "description": "The identifier of the RDS database instance; for example,
 mydbinstance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(mc|ams|awsms)-)[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]
{0,62}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DBInstanceClass": {
          "description": "The new compute and memory capacity of the DB instance, for
 example db.m4.large.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^db.[a-z0-9]+.[a-z0-9]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
        "ApplyImmediately": {
          "description": "True to apply the change immediately, false to schedule the
 change on next maintenance window. To discover your next maintenance window, check the
 details page for the instance in the RDS console.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "DBInstanceIdentifier",
          "DBInstanceClass",
          "ApplyImmediately"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBInstanceIdentifier",
        "DBInstanceClass",
        "ApplyImmediately"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-13xvbj5pqg253

Classifications:

- Management | Advanced stack components | Directory Service | Accept sharing
- Management | Directory Service | Directory | Accept sharing


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Accept Directory Sharing Request",
  "description": "Accept a directory sharing request sent from the directory owner
 account. This is run in the directory consumer account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-AcceptSharedDirectory.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AcceptSharedDirectory"
      ],
      "default": "AWSManagedServices-AcceptSharedDirectory"
    },
    "Region": {
      "description": "The AWS Region where the directory is located, in the form of us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SharedDirectoryId": {
          "description": "Identifier of the shared directory in the directory consumer
 account. This identifier is different for each directory owner account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^d-[0-9a-f]{10}$"
          },
          "maxItems": 1
        },
        "OwnerAccountId": {
```

```
          "description": "Identifier for the directory owner account that is sharing
 the directory.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SharedDirectoryId",
          "OwnerAccountId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SharedDirectoryId",
        "OwnerAccountId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-14027q0sjyt1h

Classifications:

- Deployment | Advanced stack components | EC2 stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EC2 stack",
  "description": "Use to create an Amazon Elastic Compute Cloud (EC2) instance.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "The VPC identifier (ID), in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.
 Set a Name tag to give the instance a name in the EC2 console.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
```

```
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 360
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "InstanceAmiId": {
          "description": "The AMI to use to create the EC2 instance, in the form
ami-0123abcd or ami-01234567890abcdef.",
          "type": "string",
          "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$"
        },
        "InstanceDetailedMonitoring": {
          "description": "True to enable detailed monitoring on the instance, false
to use only basic monitoring. EC2 detailed monitoring provides more frequent metrics,
published at one-minute intervals, instead of the five-minute intervals used in Amazon
EC2 basic monitoring. Detailed monitoring does incur charges. For more information,
see AWS CloudWatch documentation.",
          "type": "boolean",
          "default": false
        },
```

```
      "InstanceEBSOptimized": {
        "description": "True for the instance to be optimized for Amazon Elastic
Block Store I/O, false for it to not be. If you set this to true, choose an
InstanceType that supports EBS optimization.",
        "type": "boolean",
        "default": false
      },
      "InstanceProfile": {
        "description": "An IAM instance profile defined in your account for the EC2
instance. The default is an AWS-provided role.",
        "type": "string",
        "minLength": 1,
        "maxLength": 128,
        "pattern": "^[a-zA-Z0-9_.=@,+-]{1,128}$",
        "default": "customer-mc-ec2-instance-profile"
      },
      "InstanceRootVolumeIops": {
        "description": "The Iops to use for the root volume if volume type is io1,
io2 or gp3. If InstanceRootVolumeType is gp3, then the Iops should be between 3000 and
16000, else it should be between 100 and 64000.",
        "type": "number",
        "minimum": 100,
        "maximum": 64000,
        "default": 100
      },
      "InstanceRootVolumeName": {
        "description": "The name of the root volume to use. Defaults to /dev/xvda for
Linux, and /dev/sda for Windows.",
        "type": "string"
      },
      "InstanceRootVolumeSize": {
        "description": "The size of the root volume for the instance. Defaults to 20
GiB for Linux, and 60 GiB for Windows.",
        "type": "number",
        "minimum": 20,
        "maximum": 16000
      },
      "InstanceRootVolumeType": {
        "description": "Choose io1, io2, gp2 or gp3 for SSD-backed volumes optimized
for transactional workloads. Choose standard for HDD-backed volumes suitable for
workloads where data is infrequently accessed.",
        "type": "string",
        "enum": [
          "standard",
```

```
            "io1",
            "io2",
            "gp2",
            "gp3"
          ],
          "default": "gp3"
        },
        "InstancePrivateStaticIp": {
          "description": "The static IP address that the instance can support.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$"
        },
        "SecurityGroupIds": {
          "description": "IDs of the existing security groups to associate with the
 instance, in the form sg-0123abcd or sg-01234567890abcdef. If nothing is specified,
 the default AMS security groups will be applied.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "InstanceSubnetId": {
          "description": "The subnet that you want to launch the instance into, in the
 form subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "string",
          "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
        },
        "InstanceType": {
          "description": "The type of EC2 instance to deploy. If InstanceEBSOptimized =
 true, specify an InstanceType that supports EBS optimization.",
          "type": "string",
          "default": "t2.large"
        },
        "InstanceUserData": {
          "description": "A newline-delimited string where each line is part of the
 script to be run on boot.",
          "type": "string",
          "maxLength": 4096,
          "default": ""
        },
```

```
        "EnforceIMDSV2": {
          "description": "Set to 'false' for the instance to be launched with IMDSv1
only. Default value is 'true'. See EC2/IMDS document for more details.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "InstanceAmiId",
          "InstanceSubnetId",
          "InstanceDetailedMonitoring",
          "InstanceEBSOptimized",
          "InstanceProfile",
          "InstanceRootVolumeIops",
          "InstanceRootVolumeName",
          "InstanceRootVolumeSize",
          "InstanceRootVolumeType",
          "InstancePrivateStaticIp",
          "InstanceType",
          "InstanceUserData",
          "SecurityGroupIds",
          "EnforceIMDSV2"
        ]
      },
      "required": [
        "InstanceAmiId",
        "InstanceSubnetId",
        "EnforceIMDSV2"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
```

```
      "TimeoutInMinutes",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1404e21baa2ox

Classifications:

- Management | Custom Stack | Stack from CloudFormation Template | Approve Changeset and Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Approve ChangeSet and update CloudFormation stack",
  "description": "Approve and execute an existing ChangeSet to update a CloudFormation
 stack. This ChangeType is used primarily to approve and apply changes requested
 using the \"Update CloudFormation stack\" CT that would cause removal or replacement
 of resources, but can also be used to execute any existing ChangeSet to update
 CloudFormation stacks.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "Identifier of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "Identifier for the existing CloudFormation-based stack to be
 updated.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
```

```
    },
    "ChangeSetName": {
      "description": "Name of the ChangeSet to execute against the stack. If the stack
 update was requested using the \"Update CloudFormation stack\" CT, the ChangeSet name
 can be found in the failure reason of that RFC. You can also find the ChangeSet name
 from the ChangeSet ID which can be obtained from CloudFormation console, the ChangeSet
 ID has the format of arn:${Partition}:cloudformation:${Region}:${Account}:changeSet/
${ChangeSetName}/${Id}.",
      "type": "string",
      "pattern": "^[a-zA-Z][-a-zA-Z0-9]*$",
      "maxLength": 128
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
 the change. This does not prolong execution, but the RFC fails if the change is not
 completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 1080,
      "default": 360
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "ChangeSetName",
      "TimeoutInMinutes"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "ChangeSetName",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-14v49adibs4db

Classifications:

- Management | AMS Resource Scheduler | State | Disable


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disable AMS Resource Scheduler",
  "description": "Disable AMS Resource Scheduler in the account. This will prevent
 resources from being scheduled for automatic start or stop actions even if they are
 configured for such actions.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAMSResourceSchedulerStack-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
      ],
      "default": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
 solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SchedulingActive": {
          "description": "Specify the value: No. This explicitly requests that the
 Resource Scheduler be disabled. Default is No.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "No"
            ],
            "default": "No"
```

```
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Action": {
          "type": "string",
          "description": "(Required) The Action to be performed.",
          "enum": [
            "Update"
          ],
          "default": "Update"
        }
      },
      "metadata": {
        "ui:order": [
          "SchedulingActive",
          "Action"
        ]
      },
      "required": [
        "SchedulingActive",
        "Action"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-14yjom3kvpinu

Classifications:

- Deployment | Advanced stack components | Listener | Create (for ALB or NLB)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create ALB or NLB Listener",
  "description": "Create a listener for an Application Load Balancer (ALB) or Network
 Load Balancer (NLB). A listener is a process that checks for connection requests, the
 rules that you define for a listener determine how the load balancer routes requests
 to its registered targets.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-12345678 or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true,
      "items": {
        "type": "object",
```

```
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      }
    },
    "StackTemplateId": {
      "description": "Must be stm-u5n0r6aacdvdwthhm.",
      "type": "string",
      "enum": [
        "stm-u5n0r6aacdvdwthhm"
      ]
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
        "LoadBalancerArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the load balancer to
 associate with the listener, in the form arn:aws:elasticloadbalancing:region:account-
id:loadbalancer/load-balancer-type/load-balancer-name/load-balancer-id.",
          "pattern": "arn:aws:elasticloadbalancing:[a-z1-9\\-]{9,15}:[0-9]
{12}:loadbalancer/(net|app)/[a-zA-Z0-9\\-]{1,32}/[a-z0-9]+"
        },
        "CertificateArn": {
          "type": "string",
          "description": "The ARN of the certificate to associate with the
 listener, in the form arn:aws:acm:region:account-id:certificate/certificate-id or
 arn:aws:iam::account-id:server-certificate/certificate-name. Leave blank if Protocol
 is not HTTPS.",
          "pattern": "|(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})|(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)",
          "default": ""
        },
        "DefaultActionTargetGroupArn": {
          "type": "string",
          "description": "The ARN of the target group to which Elastic Load Balancing
 routes the traffic, in the form arn:aws:elasticloadbalancing:region:account-
id:targetgroup/target-group-name/target-group-id.",
          "pattern": "arn:aws:elasticloadbalancing:[a-z1-9\\-]{9,15}:[0-9]
{12}:targetgroup/[a-zA-Z0-9\\-]{1,32}/[a-z0-9]+"
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$|^$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer). For ALB, the supported protocols are HTTP and
 HTTPS. For NLB, the supported protocols are TCP, TLS, UDP, TCP_UDP.",
          "enum": [
            "HTTP",
            "HTTPS",
            "TCP",
            "TLS",
            "UDP",
```

```
            "TCP_UDP"
          ]
        },
        "ALBSslPolicy": {
          "type": "string",
          "description": "The ALB security policy that defines the ciphers and
protocols that the load balancer supports. Only applicable if Protocol = HTTPS.",
          "enum": [
            "",
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ],
          "default": "ELBSecurityPolicy-TLS13-1-2-2021-06"
        },
        "NLBSslPolicy": {
          "description": "The NLB security policy that defines the ciphers and
protocols that the load balancer supports. Only applicable if Protocol = TLS.",
          "type": "string",
          "enum": [
            "",
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
```

```
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ],
          "default": "ELBSecurityPolicy-TLS13-1-2-2021-06"
        },
        "AlpnPolicy": {
          "description": "The name of the Application-Layer Protocol Negotiation
(ALPN) policy that includes the protocol negotiation within the exchange of hello
messages.",
          "type": "string",
          "enum": [
            "",
            "HTTP1Only",
            "HTTP2Only",
            "HTTP2Optional",
            "HTTP2Preferred",
            "None"
          ],
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "LoadBalancerArn",
          "DefaultActionTargetGroupArn",
          "Port",
          "Protocol",
          "CertificateArn",
          "ALBSslPolicy",
          "NLBSslPolicy",
          "AlpnPolicy"
        ]
      },
      "required": [
        "LoadBalancerArn",
        "DefaultActionTargetGroupArn",
```

```
        "Port",
        "Protocol"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-15mazjj88xc69

Classifications:

- Management | Advanced stack components | EC2 instance stack | Resize

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Resize EC2 Instance",
  "description": "Resize an existing EC2 instance in your account. The state of the
 instance can be either 'running' or 'stopped'. If 'running', the instance is stopped
 during the resize operation and returned to the initial state after the resizing is
 complete. Before resizing the instance, ensure that the instance's root volume is
 not an instance store volume. We highly recommended rigorous load and performance
```

```
testing before, and after, making instance type changes, and that you also consider
the pricing changes that result when instances are resized. Please be aware that this
change may result in CloudFormation drift for any stacks that have this resource.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-ChangeInstanceType.",
     "type": "string",
     "enum": [
       "AWSManagedServices-ChangeInstanceType"
     ],
     "default": "AWSManagedServices-ChangeInstanceType"
   },
   "Region": {
     "description": "The AWS Region where the instance is, in the form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "InstanceId": {
         "description": "The ID of the instance to resize, in the form
i-12345678901234567 or i-12345678.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "InstanceType": {
         "description": "The instance type to resize to; for example, t3.xlarge
or m4.xlarge. Ensure that the instance type you select has the same underlying
hypervisor, either xen or nitro, as the instance type that you are resizing. Choosing
an instance type with a different underlying hypervisor is disallowed.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^[a-z-0-9]+\\.[a-z0-9]+$"
         },
         "minItems": 1,
         "maxItems": 1
```

```
        },
        "CreateAMIBeforeResize": {
          "description": "True to create an EC2 instance AMI as a backup before
 resizing the instance, false to not.",
          "type": "array",
          "items": {
            "type": "boolean",
            "default": false
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId",
        "InstanceType"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-16pknsfa8lul7

Classifications:

- Deployment | Managed landing zone | Management account | Create StackSets stack (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create StackSets Stack",
  "description": "Create AWS CloudFormation (CFN) StackSets stacks and deploy the stack
 instances. Use the CloudFormation StackSets feature to create stacks across multiple
 accounts.",
  "type": "object",
  "properties": {
    "CloudFormationTemplate": {
      "description": "The CFN template that you have configured to create the resources
 that you want, copy the JSON and paste it into the field. Provide a value for either
 this, or the CloudFormationTemplateS3Endpoint parameter.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?![\\s]*https?)[\\S\\s]*$",
      "maxLength": 20000
    },
    "CloudFormationTemplateS3Endpoint": {
      "description": "The S3 bucket endpoint for the CloudFormation template you want
 to use. The bucket must be in the same account that you are using, or have a presigned
 URL. Provide a value for either this, or the CloudFormationTemplate parameter.",
      "type": "string",
      "minLength": 1,
      "pattern": "^[\\s]*https?://[\\S]*[\\s]*$|^[\\s]*$",
      "maxLength": 2047
    },
    "Parameters": {
      "description": "Add up to sixty parameters (parameter name/value pairs) to
 supply alternate values for parameters in your customized CloudFormation template.
 By providing the parameters this way, you can reuse your CloudFormation template with
 different parameter values when needed and can update any parameter value with the CFN
 Update stack set (review required) change type (ct-1v9g9n30woc8h).",
      "type": "array",
      "items": {
        "type": "object",
```

```
        "properties": {
          "Name": {
            "type": "string",
            "pattern": "[A-Za-z0-9]+$"
          },
          "Value": {
            "type": "string"
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Name",
            "Value"
          ]
        },
        "required": [
          "Name",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "Description": {
      "description": "Meaningful information about the StackSets stack you are
 creating.",
      "type": "string",
      "minLength": 1,
      "maxLength": 1024
    },
    "Name": {
      "description": "A meaningful name for the StackSets stack. The name must start
 with an alphabetic character and can contain only alphanumeric characters (case-
sensitive) and hyphens.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$",
      "maxLength": 128
    },
    "OuId": {
```

```
      "description": "The ID of the AWS organizational unit for the stack instances
 being deployed. If you add a parent OU as a target, StackSets also adds any child OU
 as targets. To deploy the StackSets stack instances in all OUs, use 'all'.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^(ou-[a-z0-9]{4,32}-[a-z0-9]{8,32}|r-[a-z0-9]{4,32}|all)$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "Region": {
      "description": "The AWS Region to deploy the resources, in the form of us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the StackSets
 stack.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-|aws:))[a-zA-Z0-9 .:+=@_/-]{1,128}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-|aws:))[a-zA-Z0-9 .:+=@_/-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
```

```
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "CloudFormationTemplate",
      "CloudFormationTemplateS3Endpoint",
      "Parameters",
      "Region",
      "OuId",
      "Tags",
      "Priority"
    ]
  },
  "required": [
    "Name",
    "Description",
    "Region",
    "OuId"
  ]
}
```

Schema for Change Type ct-16xg8qguovg2w

Classifications:

- Deployment | Advanced stack components | EBS Volume | Create


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create and attach up to five EBS volumes to an instance.",
  "description": "Creates up to five EBS volumes, and attaches them to an existing EC2
 instance that you specify. Does not create a root volume.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
```

```
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-hrnfpt7l0qqumcelt",
      "type": "string",
      "enum": [
        "stm-hrnfpt7l0qqumcelt"
      ],
      "default": "stm-hrnfpt7l0qqumcelt"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 45
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AvailabilityZone": {
```

```
          "type": "string",
          "description": "The Availability Zone (AZ) to create the volume in. Must
 match the AZ of the instance ID in order to attach successfully.",
          "pattern": "^[a-z]{2}-[a-z]{4,10}-[1-9]{1}[a-z]$"
        },
        "InstanceId": {
          "type": "string",
          "description": "The instance that the created EBS volumes will be attached
 to.",
          "pattern": "^i-[0-9a-f]{8}$|^i-[0-9a-f]{17}$"
        },
        "Volume1Iops": {
          "type": "string",
          "description": "The Iops to use for Volume1 if Volume1Type is io1, io2 or
 gp3. If Volume1Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume1Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume1KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume1. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume1.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume1Name": {
          "type": "string",
          "description": "The device name for Volume1 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume1.",
          "pattern": "^/dev/sd[a-z]([2-9]|1[012345])?$|^/dev/hd[a-z]([1-9]|1[012345])?
$|^/dev/xvd[b-c][a-z]$|^/dev/xvd[b-z]$|^xvd[a-z]$|^xvd[b-c][a-z]$"
        },
        "Volume1Size": {
          "type": "string",
          "description": "The size for Volume1 in GiB. Gp2 = Min: 1 GiB, Max: 16384
 GiB. io1 = Min: 4 GiB, Max: 16384 GiB. sc1 = Min: 500 GiB, Max: 16384 GiB. st1 = Min:
 500 GiB, Max: 16384 GiB. standard = Min: 1 GiB, Max: 1024 GiB.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
```

```
        "Volume1Snapshot": {
          "type": "string",
          "description": "The snapshot identifier to create EBS Volume1. Leave blank to
 create an empty Volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume1Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume1 if Volume1Type is gp3. If
 Volume1Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume1Type": {
          "type": "string",
          "description": "The volume type for Volume1. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ],
          "default": "gp3"
        },
        "Volume2Iops": {
          "type": "string",
          "description": "The Iops to use for Volume2 if Volume2Type is io1, io2 or
 gp3. If Volume2Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume2Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume2KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume2. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume2.",
```

```
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume2Name": {
          "type": "string",
          "description": "The device name for Volume2 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume2. Leave blank to skip creation of Volume2.",
          "pattern": "^/dev/sd[a-z]([2-9]|1[012345])?$|^/dev/hd[a-z]([1-9]|1[012345])?
$|^/dev/xvd[b-c][a-z]$|^/dev/xvd[b-z]$|^xvd[a-z]$|^xvd[b-c][a-z]$|^$"
        },
        "Volume2Size": {
          "type": "string",
          "description": "The size for Volume2 in GiB. Gp2 = Min: 1 GiB, Max: 16384
 GiB. io1 = Min: 4 GiB, Max: 16384 GiB. sc1 = Min: 500 GiB, Max: 16384 GiB. st1 = Min:
 500 GiB, Max: 16384 GiB. standard = Min: 1 GiB, Max: 1024 GiB.",
          "pattern": "^$|^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5]
[0-9]{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume2Snapshot": {
          "type": "string",
          "description": "The snapshot identifier to create EBS Volume2. Leave blank to
 create an empty Volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume2Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume2 if Volume2Type is gp3. If
 Volume2Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume2Type": {
          "type": "string",
          "description": "The volume type for Volume2. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
```

```
            "st1",
            "standard"
          ],
          "default": "gp3"
        },
        "Volume3Iops": {
          "type": "string",
          "description": "The Iops to use for Volume3 if Volume3Type is io1, io2 or
 gp3. If Volume3Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume3Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume3KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume3. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume3.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume3Name": {
          "type": "string",
          "description": "The device name for Volume3 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume3. Leave blank to skip creation of Volume3.",
          "pattern": "^/dev/sd[a-z]([2-9]|1[012345])?$|^/dev/hd[a-z]([1-9]|1[012345])?
$|^/dev/xvd[b-c][a-z]$|^/dev/xvd[b-z]$|^xvd[a-z]$|^xvd[b-c][a-z]$|^$"
        },
        "Volume3Size": {
          "type": "string",
          "description": "The size for Volume3 in GiB. Gp2 = Min: 1 GiB, Max: 16384
 GiB. io1 = Min: 4 GiB, Max: 16384 GiB. sc1 = Min: 500 GiB, Max: 16384 GiB. st1 = Min:
 500 GiB, Max: 16384 GiB. standard = Min: 1 GiB, Max: 1024 GiB.",
          "pattern": "^$|^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5]
[0-9]{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume3Snapshot": {
          "type": "string",
          "description": "The snapshot identifier to create EBS Volume3. Leave blank to
 create an empty Volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
```

```
        "Volume3Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume3 if Volume3Type is gp3. If
 Volume3Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume3Type": {
          "type": "string",
          "description": "The volume type for Volume3. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ],
          "default": "gp3"
        },
        "Volume4Iops": {
          "type": "string",
          "description": "The Iops to use for Volume4 if Volume4Type is io1, io2 or
 gp3. If Volume4Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume4Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume4KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume4. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume4.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume4Name": {
          "type": "string",
```

```
          "description": "The device name for Volume4 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume4. Leave blank to skip creation of Volume4.",
          "pattern": "^/dev/sd[a-z]([2-9]|1[012345])?$|^/dev/hd[a-z]([1-9]|1[012345])?
$|^/dev/xvd[b-c][a-z]$|^/dev/xvd[b-z]$|^xvd[a-z]$|^xvd[b-c][a-z]$|^$"
        },
        "Volume4Size": {
          "type": "string",
          "description": "The size for Volume4 in GiB. Gp2 = Min: 1 GiB, Max: 16384
 GiB. io1 = Min: 4 GiB, Max: 16384 GiB. sc1 = Min: 500 GiB, Max: 16384 GiB. st1 = Min:
 500 GiB, Max: 16384 GiB. standard = Min: 1 GiB, Max: 1024 GiB.",
          "pattern": "^$|^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5]
[0-9]{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume4Snapshot": {
          "type": "string",
          "description": "The snapshot identifier to create EBS Volume4. Leave blank to
 create an empty Volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume4Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume4 if Volume4Type is gp3. If
 Volume4Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume4Type": {
          "type": "string",
          "description": "The volume type for Volume4. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ],
          "default": "gp3"
        },
```

```
        "Volume5Iops": {
          "type": "string",
          "description": "The Iops to use for Volume5 if Volume5Type is io1, io2 or
 gp3. If Volume5Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume5Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume5KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume5. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume5.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume5Name": {
          "type": "string",
          "description": "The device name for Volume5 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume5. Leave blank to skip creation of Volume5.",
          "pattern": "^/dev/sd[a-z]([2-9]|1[012345])?$|^/dev/hd[a-z]([1-9]|1[012345])?
$|^/dev/xvd[b-c][a-z]$|^/dev/xvd[b-z]$|^xvd[a-z]$|^xvd[b-c][a-z]$|^$"
        },
        "Volume5Size": {
          "type": "string",
          "description": "The size for Volume5 in GiB. Gp2 = Min: 1 GiB, Max: 16384
 GiB. io1 = Min: 4 GiB, Max: 16384 GiB. sc1 = Min: 500 GiB, Max: 16384 GiB. st1 = Min:
 500 GiB, Max: 16384 GiB. standard = Min: 1 GiB, Max: 1024 GiB.",
          "pattern": "^$|^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5]
[0-9]{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume5Snapshot": {
          "type": "string",
          "description": "The snapshot identifier to create EBS Volume5. Leave blank to
 create an empty Volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume5Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume5 if Volume5Type is gp3.
 If Volume5Type is not gp3, any value provided here is ignored. Default is 125. The
 Throughput should be between 125 and 1000.",
```

```
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume5Type": {
          "type": "string",
          "description": "The volume type for Volume5. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ],
          "default": "gp3"
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "AvailabilityZone",
          "Volume1Name",
          "Volume1Size",
          "Volume1Type",
          "Volume1Iops",
          "Volume1Throughput",
          "Volume1KmsKeyId",
          "Volume1Snapshot",
          "Volume2Name",
          "Volume2Size",
          "Volume2Type",
          "Volume2Iops",
          "Volume2Throughput",
          "Volume2KmsKeyId",
          "Volume2Snapshot",
          "Volume3Name",
          "Volume3Size",
          "Volume3Type",
          "Volume3Iops",
          "Volume3Throughput",
          "Volume3KmsKeyId",
```

```
          "Volume3Snapshot",
          "Volume4Name",
          "Volume4Size",
          "Volume4Type",
          "Volume4Iops",
          "Volume4Throughput",
          "Volume4KmsKeyId",
          "Volume4Snapshot",
          "Volume5Name",
          "Volume5Size",
          "Volume5Type",
          "Volume5Iops",
          "Volume5Throughput",
          "Volume5KmsKeyId",
          "Volume5Snapshot"
        ]
      },
      "required": [
        "InstanceId",
        "AvailabilityZone",
        "Volume1Name",
        "Volume1Size"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
```

```
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1706xvvk6j9hf

Classifications:

- Management | Managed account | Automated IAM provisioning with read-write permissions | Enable (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Automated IAM Provisioning",
  "description": "Enable Automated IAM provisioning with read-write
 permissions in the account used to submit this CT. Once enabled, a new role
 'AWSManagedServicesIAMProvisionAdminRole' is created in that account. Additionally,
 you can use three related change types (ct-1n9gfnog5x7fl, ct-1e0xmuy1diafq,
 ct-17cj84y7632o6) to create, update, or delete IAM roles and policies using Automated
 IAM provisioning with read-write permissions, which employs an automated review
 process with a predefined set of rules for IAM and AMS. Before using, we recommend
 a good familiarity with IAM rules. To confirm that an account has Automated IAM
 provisioning enabled, look for the IAM role 'AWSManagedServicesIAMProvisionAdminRole'
 in the IAM console for that account.",
  "type": "object",
  "properties": {
    "SAMLIdentityProviderArns": {
      "description": "Comma-separated list of the SAML identity provider (IdP) ARNs to
 assume the Automated IAM provisioning role. You must set at least one provider, using
 either this parameter or IamEntityArns.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^arn:aws:iam::\\d{12}:saml-provider\\/[\\w._+=,@-]{1,128}$"
      },
      "uniqueItems": true
    },
    "IamEntityArns": {
      "description": "Comma-separated list of ARNs of the IAM entities to assume the
 Automated IAM provisioning role. You must set at least one IAM principal, using either
 this parameter or SAMLIdentityProviderArns.",
```

```
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^arn:aws:iam::\\d{12}:role\\/[\\w+=,.@-]{1,64}$"
      },
      "uniqueItems": true
    },
    "CustomerCustomDenyActionsList1": {
      "description": "Comma-separated list of actions to be denied in IAM roles created
 by the Automated IAM provisioning role.",
      "type": "string",
      "pattern": "^[a-z0-9-]+:[A-Za-z0-9*-]+(?:,[a-z0-9-]+:[A-Za-z0-9*-]+)*$",
      "maxLength": 4096
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "default": "High",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "SAMLIdentityProviderArns",
      "IamEntityArns",
      "CustomerCustomDenyActionsList1",
      "Priority"
    ]
  }
}
```

Schema for Change Type ct-17cj84y7632o6

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Delete entity or policy (read-write permissions)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Entity or Policy (read-write permissions)",
  "description": "Delete Identity and Access Management (IAM) role or policy created
 with change type ct-1n9gfnog5x7fl.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAutomatedIAMProvisioningDelete-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin"
      ],
      "default": "AWSManagedServices-HandleAutomatedIAMProvisioningDelete-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
```

```
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RoleName": {
          "description": "A list of up to five IAM role names to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$"
          },
          "minItems": 0,
          "maxItems": 5,
          "uniqueItems": true
        },
        "ManagedPolicyName": {
          "description": "A list of up to five IAM customer managed policy names to
delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,128}$"
          },
          "minItems": 0,
          "maxItems": 5,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "RoleName",
          "ManagedPolicyName"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
```

```
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-17vnu10suy631

Classifications:

- Deployment | Advanced stack components | Cache (ElastiCache Redis) stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Cache (ElastiCache Redis) stack",
  "description": "Use to create an Amazon ElastiCache cluster (one or more cache nodes)
 that uses the Redis engine.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the vpc to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-sfpo2o00000000000.",
      "type": "string",
      "enum": [
        "stm-sfpo2o00000000000"
```

```
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
```

```
      "maximum": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "ElastiCacheAutoMinorVersionUpgrade": {
          "description": "True for minor engine upgrades to be applied automatically
to the cache cluster during the specified ElastiCachePreferredMaintenanceWindow, false
for the upgrades to not be applied automatically. Default is true.",
          "type": "boolean",
          "default": true
        },
        "ElastiCacheBackupSnapshotRetentionLimit": {
          "description": "The number of days for which Redis retains automatic
snapshots before deleting them.",
          "type": "number",
          "default": 7,
          "minimum": 0,
          "maximum": 30
        },
        "ElastiCacheClusterName": {
          "description": "A name for the cache cluster.",
          "type": "string",
          "minLength": 1,
          "maxLength": 20,
          "pattern": "^[a-zA-Z][a-zA-Z0-9-]{0,18}[a-zA-Z0-9]$|^[a-zA-Z]$"
        },
        "ElastiCacheCPUThresholdAlarmOverride": {
          "description": "The value for CPUUtilization metric maximum threshold if the
automatically derived one from the instance type needs to be overridden.",
          "type": "number",
          "default": 0,
          "minimum": 0,
          "maximum": 100
        },
        "ElastiCacheEnableBackup": {
          "description": "True to enable periodic backups for the cache cluster, false
to not. Default is false.",
          "type": "boolean",
          "default": false
        },
        "ElastiCacheEngine": {
          "description": "Must be redis.",
```

```
          "type": "string",
          "enum": [
            "redis"
          ]
        },
        "ElastiCacheEngineVersion": {
          "description": "The version of the Redis cache engine to be used for the
 cluster.",
          "type": "string"
        },
        "ElastiCacheInstanceType": {
          "description": "The compute and memory capacity of nodes in the Redis cache
 cluster.",
          "type": "string",
          "default": "cache.t3.micro"
        },
        "ElastiCachePort": {
          "description": "The port number on which each of the cache nodes will accept
 connections.",
          "type": "number",
          "minimum": 0,
          "maximum": 65535,
          "default": 6379
        },
        "ElastiCachePreferredBackupWindow": {
          "description": "The daily time range (in UTC) during which Redis will
 begin taking a daily snapshot of your node group. For example, you can specify
 05:00-09:00.",
          "type": "string",
          "default": "22:00-23:00",
          "pattern": "^(?:[0-1][0-9]|2[0-3]):[0-5][0-9]-(?:[0-1][0-9]|2[0-3]):[0-5]
[0-9]$"
        },
        "ElastiCachePreferredMaintenanceWindow": {
          "description": "The weekly time range (in UTC) during which system
 maintenance can occur. For example, you can specify: sun:02:00-sun:04:00.",
          "type": "string",
          "pattern": "^(?:sun|mon|tue|wed|thu|fri|sat):(?:[0-1][0-9]|2[0-3]):[0-5]
[0-9]-(?:sun|mon|tue|wed|thu|fri|sat):(?:[0-1][0-9]|2[0-3]):[0-5][0-9]$"
        },
        "ElastiCacheSnapshotArns": {
          "description": "The ARN of the snapshot file that you want to use to seed a
 new Redis cache cluster.",
          "type": "string",
```

```
          "minLength": 16,
          "pattern": "^arn:aws:s3:"
        },
        "ElastiCacheSnapshotName": {
          "description": "The name of a snapshot from which to restore data into the
new Redis cache cluster.",
          "type": "string"
        },
        "ElastiCacheSubnetGroup": {
          "description": "The subnet group name that you want to associate with the
cluster.",
          "type": "string",
          "minLength": 1,
          "maxLength": 255,
          "pattern": "^[a-z0-9-]{1,255}$"
        },
        "ElastiCacheSubnetIds": {
          "description": "One or more subnet IDs for the cache cluster, in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1
        },
        "SecurityGroups": {
          "description": "One or more VPC security groups that you want to associate
with the cluster, in the form sg-0123abcd or sg-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 1
        }
      },
      "additionalProperties": false,
      "required": [
        "ElastiCacheClusterName",
        "ElastiCacheEngine",
        "ElastiCacheSubnetIds"
      ]
    }
```

```
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "Parameters",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-17w6f6kzf6w51

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create DB subnet group

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS DB subnet group",
  "description": "Create a Relational Database Service (RDS) database (DB) subnet group
 to be used with a specified RDS DB.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
```

```
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-iutsfv5ci7suupr86",
      "type": "string",
      "enum": [
        "stm-iutsfv5ci7suupr86"
      ],
      "default": "stm-iutsfv5ci7suupr86"
```

```
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBSubnetGroupName": {
          "type": "string",
          "description": "The name of your DB subnet group. Must contain 1 to 255
alphanumeric characters including period, underscore, and hyphen; and must be unique
per account per region. Cannot be named \"default.\"",
          "pattern": "^(?!default$)[a-zA-Z0-9._-]{1,255}$"
        },
        "DBSubnetGroupDescription": {
          "type": "string",
          "description": "A description to help identify your DB subnet group. If blank
the subnet group name is used.",
          "default": ""
        },
        "SubnetIds": {
          "type": "array",
          "minItems": 2,
          "uniqueItems": true,
          "description": "Two or more subnet IDs to include in the DB subnet group,
in the form subnet-0123abcd or subnet-01234567890abcdef, spanning at least two
Availability Zones.",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          }
        }
      },
      "metadata": {
        "ui:order": [
          "DBSubnetGroupName",
          "DBSubnetGroupDescription",
          "SubnetIds"
```

```
        ]
      },
      "required": [
        "DBSubnetGroupName",
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1895yr1p87noq

Classifications:

- Management | AWS Backup | Backup job | Stop

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop Backup Job",
  "description": "Stop an AWS Backup service running, or scheduled, backup job.",
  "type": "object",
```

```
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StopBackupJob.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StopBackupJob"
      ],
      "default": "AWSManagedServices-StopBackupJob"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BackupJobId": {
          "description": "The ID of the AWS Backup target job.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-
[a-fA-F0-9]{12}){1}$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "BackupJobId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BackupJobId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
```

```
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-18fzkt86jmw1s

Classifications:

- Deployment | Patching | SSM patch baseline | Create (Amazon Linux 2)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Baseline (Amazon Linux 2)",
  "description": "Create an AWS Systems Manager (SSM) patch baseline to define which
 patches are approved for installation on your instances for Amazon Linux 2 OS.
 Specify existing instance \"Patch Group\" tag values for the patch baseline. The patch
 baseline is an SSM resource that you can manage with the SSM console.",
  "additionalProperties": false,
  "properties": {
    "ApprovalRules": {
      "description": "Create auto-approval rules to specify that certain types of
 operating system patches are approved automatically.",
      "items": {
        "additionalProperties": false,
        "properties": {
          "ApproveAfterDays": {
            "default": 7,
            "description": "The number of days to wait after a patch is released before
 approving patches automatically.",
            "maximum": 100,
            "minimum": 0,
            "type": "integer"
          },
          "Classification": {
```

```
            "description": "The Classification of the patches to be selected. Allowed
values are \"All\", \"Bugfix\", \"Enhancement\", \"Newpackage\", \"Recommended\" and
\"Security\".",
            "items": {
              "enum": [
                "All",
                "Bugfix",
                "Enhancement",
                "Newpackage",
                "Recommended",
                "Security"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          },
          "Severity": {
            "description": "The severity of the patches to be selected. Allowed values
are \"All\", \"Critical\", \"Important\", \"Low\" and \"Medium\".",
            "items": {
              "enum": [
                "All",
                "Critical",
                "Important",
                "Low",
                "Medium"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          }
        },
        "metadata": {
          "ui:order": [
            "Severity",
            "Classification",
            "ApproveAfterDays"
          ]
        },
        "required": [
          "ApproveAfterDays"
        ],
```

```
        "type": "object"
      },
      "maxItems": 10,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "ApprovedPatches": {
      "description": "The list of patches to approve explicitly.",
      "items": {
        "type": "string",
        "maxLength": 100,
        "minLength": 1
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Description": {
      "description": "A meaningful description for this patch baseline.",
      "maxLength": 500,
      "minLength": 1,
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch baseline.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "OperatingSystem": {
      "default": "Amazon Linux 2",
      "description": "The operating system of instances to which this baseline is
 applied.",
      "enum": [
        "Amazon Linux 2"
      ],
      "type": "string"
    },
    "PatchGroupTagValues": {
```

```
      "description": "A list of the values of your \"Patch Group\" tags on the
instances you want patched; the values for up to twenty-five \"Patch Group\" tags can
be provided. Instances with those values are associated with this patch baseline.",
      "items": {
        "maxLength": 256,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 25,
      "minItems": 1,
      "type": "array",
      "uniqueItems": true
    },
    "RejectedPatches": {
      "description": "The list of patches to reject explicitly.",
      "items": {
        "maxLength": 100,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the SSM patch
baseline resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
```

```
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "OperatingSystem",
      "Name",
      "Description",
      "PatchGroupTagValues",
      "ApprovalRules",
      "ApprovedPatches",
      "RejectedPatches",
      "Tags"
    ]
  },
  "required": [
    "Name",
    "PatchGroupTagValues",
    "OperatingSystem"
  ],
  "type": "object"
}
```

Schema for Change Type ct-18r16ldqil6w9

Classifications:

- Management | Advanced stack components | Security group | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Security Groups",
  "description": "Delete up to 20 security groups. Note: Only security groups with no
 dependencies are deleted and security groups with dependencies are not deleted. This
 change type does not require a review and can be used instead of the manual, review
 required, change type (ct-3cp96z7r065e4).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteSecurityGroups.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteSecurityGroups"
      ],
      "default": "AWSManagedServices-DeleteSecurityGroups"
    },
    "Region": {
      "description": "The AWS Region in which the security group is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityGroupIds": {
          "description": "A list of up to 20 security group IDs to be deleted.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 20,
          "uniqueItems": true
        },
        "ForceDelete": {
          "description": "True to delete the security groups with only Auto Scaling
 launch template or launch configuration dependencies, or false if not. Default is
 false, and only security groups with no dependencies are deleted. Note: Auto Scaling
 Group or EC2 instances using Launch Templates or Launch Configurations with deleted
 security groups cannot be launched.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "default": "false",
            "enum": [
              "true",
              "false"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SecurityGroupIds",
          "ForceDelete"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SecurityGroupIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1962s5oczal9z

Classifications:

- Management | Advanced stack components | Bastions | Update instance or session counts (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Instance or Session Counts",
  "description": "Update the number of RDP and SSH Bastion instances. Optionally update
 the session count of RDP Bastions.",
  "type": "object",
  "properties": {
    "BastionType": {
      "description": "The bastion type to update, this determines which parameters
 are applicable. RDP Bastion type applies to all of the parameters. SSH Bastion type
 applies to only the ASGMaxCount, ASGMinCount, ASGDesiredCount parameters.",
      "type": "string",
      "enum": [
        "RDP Bastion",
        "SSH Bastion"
      ]
    },
    "RDPBastionDesiredMaximumSessions": {
      "description": "RDP bastion desired maximum number of sessions.",
      "type": "integer"
    },
    "RDPBastionDesiredMinimumSessions": {
      "description": "RDP bastion desired minimum number of sessions.",
      "type": "integer"
    },
    "ASGMaxCount": {
      "description": "The maximum number of bastion instances to run in the bastion
 ASG.",
      "type": "integer"
    },
    "ASGMinCount": {
      "description": "The minimum number of bastion instances to run in the bastion
 ASG.",
      "type": "integer"
    },
```

```
    "ASGDesiredCount": {
      "description": "The preferred number of bastion instances to run in the bastion
 ASG.",
      "minimum": 1,
      "type": "integer"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "BastionType",
      "RDPBastionDesiredMaximumSessions",
      "RDPBastionDesiredMinimumSessions",
      "ASGMaxCount",
      "ASGMinCount",
      "ASGDesiredCount",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "BastionType"
  ]
}
```

Schema for Change Type ct-1976sir132k22

Classifications:

- Management | AMS Resource Scheduler | Period | Add

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
 "name": "Add Resource Scheduler Period",
 "description": "Add a new period to use with AMS Resource Scheduler. Periods are used
in schedules to precisely define when a resource should run.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-AddOrUpdatePeriod.",
     "type": "string",
     "enum": [
       "AWSManagedServices-AddOrUpdatePeriod"
     ],
     "default": "AWSManagedServices-AddOrUpdatePeriod"
   },
   "Region": {
     "description": "The AWS Region of the account where the AMS Resource Scheduler
solution is, in the form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "Action": {
         "description": "Specify the value: add. This explicitly requests that the
Resource Scheduler period be added. The option cannot be left blank; it must be
add.",
         "type": "array",
         "items": {
           "type": "string",
           "enum": [
             "add"
           ],
           "default": "add"
         },
         "maxItems": 1,
         "minItems": 1
       },
       "Name": {
         "description": "A meaningful name for the period. The name must be unique for
this account.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "(?!^[-_, +=.:#/])^[A-Za-z0-9-_, +=.:#/]{1,64}$"
```

```
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Description": {
          "description": "A meaningful description for the period.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,1000}$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "BeginTime": {
          "description": "The time, in HH:MM format, a resource starts under this
period.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((?:[01]\\d|2[0-3]):[0-5]\\d)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "EndTime": {
          "description": "The time, in HH:MM format, a resource stops under this
period.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((?:[01]\\d|2[0-3]):[0-5]\\d)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Months": {
          "description": "Enter a comma-delimited list of months (e.g. jan, feb), a
hyphenated range of months (e.g. jan-dec), or every n-th month (e.g. jan/3 for every
3rd month starting from jan) during which the resource runs. Abbreviated month names
(e.g. jan, feb, march) and numbers (1, 2, 12) are supported.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,-/]*)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "MonthDays": {
          "description": "Enter a comma-delimited list of days of the month (e.g. 1,
5, 15), a hyphenated range of days (e.g. 1-15), every n-th day of the month (e.g 1/7
for every 7th day starting on the 1st) or every n-th day day of the month in a range
( e.g. 1-15/2 for every other day from 1st to the 15th), the last day of the month
(specify L), or the nearest weekday to a specific date (specify W e.g. 15W) during
which the resource runs.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,-/]*)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "WeekDays": {
          "description": "Enter a comma-delimited list of days of the week (e.g.
Mon, Wed, Fri), a range of days of the week (e.g. Mon-Thu), or n-th occurrence
of a weekday in the month (e.g Mon#1 or 0#1 for first Monday of the month) during
which the resource runs. Enter a day and L ro run a resource on the last occurrence
of that weekday in the month (e.g. friL or 4L to run on the last Friday of the
month). Abbreviated week day names (e.g. Sun, Mon, Thu), and numbers (0, 1, 3), are
supported.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,#-/]*)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Action",
          "Name",
          "Description",
          "BeginTime",
          "EndTime",
```

```
          "Months",
          "MonthDays",
          "WeekDays"
        ]
      },
      "required": [
        "Action",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-199h35t7uz6jl

Classifications:

- Management | Access | Stack read-only access | Grant

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Grant Stack Read-Only access",
  "description": "Request Read-Only access for one or more users for one or more
 stacks. The maximum access time is 12 hours.",
  "type": "object",
  "properties": {
    "DomainFQDN": {
      "description": "The FQDN for the user accounts to grant access to.",
```

```
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "StackIds": {
      "description": "A minimum of one stack ID is required.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^stack-[a-z0-9]{17}$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "TimeRequestedInHours": {
      "description": "The amount of time, in hours, requested for access to the
instance. Access is terminated after this time.",
      "type": "integer",
      "minimum": 1,
      "default": 1
    },
    "Usernames": {
      "description": "One or more Active Directory user names used to grant access.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "VpcId": {
      "description": "The ID of the VPC that contains the stacks where access is
required, in the form of vpc-12345678 or vpc-1234567890abcdef0.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackIds",
      "Usernames",
      "DomainFQDN",
      "TimeRequestedInHours"
```

```
    ]
  },
  "additionalProperties": false,
  "required": [
    "DomainFQDN",
    "StackIds",
    "Usernames",
    "VpcId"
  ]
}
```

Schema for Change Type ct-19f40lfm5umy8

Classifications:

- Management | Advanced stack components | EC2 instance stack | Gather log4j information

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Gather Log4j Information",
  "description": "Generates a report identifying Log4j2 occurrences on the specified
 EC2 instances. This is a best-effort report and some occurrences may go undetected
 from the report.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-GatherLog4jInformation.",
      "type": "string",
      "enum": [
        "AWSManagedServices-GatherLog4jInformation"
      ],
      "default": "AWSManagedServices-GatherLog4jInformation"
    },
    "Region": {
      "description": "The AWS Region in which the EC2 instances are located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
      "S3Bucket": {
        "description": "The name of the S3 bucket to upload the results to, in the
form s3://bucket-name.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^s3://.+$"
        },
        "minItems": 1,
        "maxItems": 1
      }
    },
    "metadata": {
      "ui:order": [
        "S3Bucket"
      ]
    },
    "additionalProperties": false
  },
  "TargetParameterName": {
    "description": "Must be InstanceId.",
    "type": "string",
    "enum": [
      "InstanceId"
    ],
    "default": "InstanceId"
  },
  "Targets": {
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "Key": {
          "description": "The criteria for targeting resources. To target
all instances in the AWS Region, use AWS::EC2::Instance. To target specific
instances, use ParameterValues and specify instance IDs for the Values. Default is
AWS::EC2::Instance.",
          "type": "string",
          "enum": [
            "AWS::EC2::Instance",
            "ParameterValues"
          ],
          "default": "AWS::EC2::Instance"
        },
```

```
        "Values": {
          "description": "Values for specified criteria. For Key=AWS::EC2::Instance,
use asterisk (*). For Key=ParameterValues, enter up to fifty instance IDs. Default is
asterisk (*).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[0-9a-f]{8}$|^i-[0-9a-f]{17}|\\*",
            "default": "*"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Key",
          "Values"
        ]
      },
      "required": [
        "Key",
        "Values"
      ]
    },
    "minItems": 1,
    "maxItems": 1
  },
  "MaxConcurrency": {
    "description": "The maximum number of targets allowed to run this task in
parallel. You can specify a number, such as 10, or a percentage, such as 10%. The
default value is 50.",
    "type": "string",
    "pattern": "^([1-9][0-9]*|[1-9][0-9]%|[1-9]%|100%)$",
    "default": "50"
  },
  "MaxErrors": {
    "description": "The number of errors that are allowed before the system stops
running the task on additional targets. You can specify either an absolute number
of errors, for example 10, or a percentage of the target set, for example 10%. The
default value is 100%.",
    "type": "string",
```

```
      "pattern": "^([1-9][0-9]*|[1-9][0-9]%|[0-9]%|100%)$",
      "default": "100%"
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters",
      "TargetParameterName",
      "Targets",
      "MaxConcurrency",
      "MaxErrors"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters",
    "TargetParameterName",
    "Targets",
    "MaxConcurrency",
    "MaxErrors"
  ]
}
```

Schema for Change Type ct-19fdy7np55xiu

Classifications:

- Deployment | Advanced stack components | RDS snapshot | Copy (for Aurora)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Copy RDS DB Cluster Snapshot",
  "description": "Create a copy of an Amazon Relational Database Service (Amazon RDS)
 DB Cluster snapshot. If you are copying a snapshot shared from another AWS account, it
 must be located in the same AWS Region as the specified DocumentName.",
  "type": "object",
  "properties": {
    "DocumentName": {
```

```
      "description": "Must be AWSManagedServices-CopyDBClusterSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CopyDBClusterSnapshot"
      ],
      "default": "AWSManagedServices-CopyDBClusterSnapshot"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}|^$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SourceDBClusterSnapshotARN": {
          "description": "The Amazon Resource Name (ARN) of the DB Cluster snapshot to
 be copied.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:[a-z0-9-]+:[0-9]{12}:cluster-
snapshot:[a-zA-Z][a-zA-Z0-9-:]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TargetDBClusterSnapshotIdentifier": {
          "description": "The target DB cluster snapshot identifier.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KmsKeyId": {
          "description": "An AWS Key Management Service (KMS) key to encrypt the DB
 snapshot with, either the KMS key ARN or the KMS key identifier. Leave blank if the
 source snapshot is unencrypted.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/)
{0,1}([a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|mrk-[0-9a-f]{32}$)|
^$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "SourceRegion": {
          "description": "The AWS Region where the source snapshot is located. Leave
 blank if the source snapshot is located in the same AWS Region as the specified
 DocumentName.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}|^$"
          },
          "minItems": 0,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SourceDBClusterSnapshotARN",
          "TargetDBClusterSnapshotIdentifier",
          "KmsKeyId",
          "SourceRegion"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SourceDBClusterSnapshotARN",
        "TargetDBClusterSnapshotIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
```

```
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1a1zzgi2nb83d

Classifications:

- Management | Advanced stack components | Application Load Balancer | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Application Load Balancer",
  "description": "Update the properties of an existing AWS Application Load Balancer
 (ALB) that was created by version 3.0 CT: ct-111r1yayblnw4.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the Application Load Balancer that you are
 updating, in the form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "LoadBalancerSecurityGroups": {
          "description": "A list of security groups to associate with the load
 balancer. Please note that changing this value during an update does not append to
 the existing security groups associated with the load balancer. Include all required
 security groups when modifying this value.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "uniqueItems": true
        },
        "LoadBalancerSubnetIds": {
          "description": "A list of subnet IDs to replace the currently used
subnets. If you update the LoadBalancerSubnetIds, specify subnets from at least two
Availability Zones. For an internet-facing load balancer provide public subnet IDs,
for an internal load balancer we recommend private subnet IDs.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "uniqueItems": true
        },
        "LoadBalancerDeletionProtection": {
          "type": "string",
          "description": "True to enable deletion protection, false to not. Default is
false.",
          "enum": [
            "true",
            "false"
          ]
        },
        "LoadBalancerIdleTimeout": {
          "type": "string",
          "description": "How long the load balancer front-end connection (client to
load balancer) can be idle (not receiving data) before the connection is automatically
closed.",
          "pattern": "^([1-9][0-9]{0,2}|[1-3][0-9]{3}|4000)$"
        },
        "Listener1Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
external incoming traffic.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "Listener1Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
connections (client to load balancer). The supported protocols are HTTP and HTTPS.",
          "enum": [
            "HTTP",
```

```
            "HTTPS"
          ]
        },
        "Listener1SSLCertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the certificate to
 associate with the listener, in the form arn:aws:acm:region:account-id:certificate/
certificate-id or arn:aws:iam::account-id:server-certificate/certificate-name. Leave
 blank if Protocol is not HTTPS.",
          "pattern": "^$|^(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})$|^(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$"
        },
        "Listener1SSLPolicy": {
          "type": "string",
          "description": "The security policy that defines the ciphers and protocols
 that the load balancer supports. Use only if Protocol = HTTPS. See AWS documentation
 for ALBs for details on default AWS security policies.",
          "enum": [
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ]
        },
        "Listener2Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$|^$"
```

```
        },
        "Listener2Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer). The supported protocols are HTTP and HTTPS.",
          "pattern": "^$|^(HTTP|HTTPS)$"
        },
        "Listener2SSLCertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the certificate to
 associate with the listener, in the form arn:aws:acm:region:account-id:certificate/
certificate-id or arn:aws:iam::account-id:server-certificate/certificate-name. Leave
 blank if Protocol is not HTTPS.",
          "pattern": "^$|^(arn:aws:acm:[a-z1-9\\-]{9,15}:[0-9]{12}:certificate/[a-z0-9]
{8}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{4}-[a-z0-9]{12})$|^(arn:aws:iam::[0-9]{12}:server-
certificate/[\\w+=,.@-]+)$"
        },
        "Listener2SSLPolicy": {
          "type": "string",
          "description": "The security policy that defines the ciphers and protocols
 that the load balancer supports. Use only if Protocol = HTTPS. See AWS documentation
 for ALBs for details on default AWS security policies.",
          "enum": [
            "ELBSecurityPolicy-TLS13-1-2-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Res-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext1-2021-06",
            "ELBSecurityPolicy-TLS13-1-2-Ext2-2021-06",
            "ELBSecurityPolicy-TLS13-1-1-2021-06",
            "ELBSecurityPolicy-TLS13-1-0-2021-06",
            "ELBSecurityPolicy-TLS13-1-3-2021-06",
            "ELBSecurityPolicy-FS-1-2-Res-2020-10",
            "ELBSecurityPolicy-FS-1-2-Res-2019-08",
            "ELBSecurityPolicy-FS-1-2-2019-08",
            "ELBSecurityPolicy-FS-1-1-2019-08",
            "ELBSecurityPolicy-FS-2018-06",
            "ELBSecurityPolicy-TLS-1-2-Ext-2018-06",
            "ELBSecurityPolicy-TLS-1-2-2017-01",
            "ELBSecurityPolicy-TLS-1-1-2017-01",
            "ELBSecurityPolicy-2016-08",
            "ELBSecurityPolicy-TLS-1-0-2015-04",
            "ELBSecurityPolicy-2015-05"
          ]
        },
        "TargetGroupHealthCheckInterval": {
```

```
          "type": "string",
          "description": "The approximate amount of time, in seconds, between health
 checks of an individual target. The range is 5 to 300 seconds.",
          "pattern": "^([5-9]|[1-8][0-9]|9[0-9]|[12][0-9]{2}|300)$"
        },
        "TargetGroupHealthCheckPath": {
          "type": "string",
          "description": "The ping path destination where Elastic Load Balancing sends
 health check requests.",
          "pattern": "^(/?[a-z0-9\\-._~%!$&'()*+,;=@]+(/[a-z0-9\\-._~%!$&'()*
+,;=:@]+)*/?|/){1,1024}$"
        },
        "TargetGroupHealthCheckPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health
 checks on targets. The default is traffic-port, which is the port on which each target
 receives traffic from the load balancer.",
          "pattern": "^$|^([0-9]{1,5})$"
        },
        "TargetGroupHealthCheckProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS"
          ]
        },
        "TargetGroupHealthCheckTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, to wait for a response to
 a health check. Must be less than the value for HealthCheckInterval. The supported
 values are 2 seconds to 60 seconds.",
          "pattern": "^(60|[1-5]{1}[0-9]{1}|[2-9]{1})$"
        },
        "TargetGroupHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health probe successes required
 before moving the instance to the Healthy state.",
          "pattern": "^([2-9]{1}|10)$"
        },
        "TargetGroupUnhealthyThreshold": {
          "type": "string",
```

```
          "description": "The number of consecutive health probe failures required
 before moving the instance to the Unhealthy state.",
          "pattern": "^([2-9]{1}|10)$"
        },
        "TargetGroupValidHTTPCode": {
          "type": "string",
          "description": "The HTTP codes that a healthy target application server must
 use in response to a health check. You can specify multiple values such as 200,202, or
 a range of values such as 200-499. Only applicable if HealthCheckTargetProtocol = HTTP
 or HTTPS.",
          "pattern": "^(([2-4]{1}[0-9]{2}($|-|,))+)$"
        },
        "TargetGroupDeregistrationDelayTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing
 to wait before changing the state of a deregistering target from draining to unused.
 Valid value ranges from 0 to 3600.",
          "pattern": "^(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})$"
        },
        "TargetGroupSlowStartDuration": {
          "type": "string",
          "description": "The time period, in the range 30-900 seconds, during which
 the load balancer sends a newly registered target a linearly-increasing share of the
 target group traffic",
          "pattern": "^([3-9]{1}[0-9]{1}|[1-8]{1}[0-9]{2}|900|0)$|^$"
        },
        "TargetGroupCookieExpirationPeriod": {
          "type": "string",
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "pattern": "^([1-9]{1}[0-9]{0,4}|[1-5]{1}[0-9]{5}|60[0-3]{1}[0-9]{3}|604[0-7]
{1}[0-9]{2}|604800)$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "LoadBalancerSecurityGroups",
          "LoadBalancerSubnetIds",
          "LoadBalancerDeletionProtection",
          "LoadBalancerIdleTimeout",
          "Listener1Port",
          "Listener1Protocol",
          "Listener1SSLCertificateArn",
```

```
          "Listener1SSLPolicy",
          "Listener2Port",
          "Listener2Protocol",
          "Listener2SSLCertificateArn",
          "Listener2SSLPolicy",
          "TargetGroupHealthCheckInterval",
          "TargetGroupHealthCheckPath",
          "TargetGroupHealthCheckPort",
          "TargetGroupHealthCheckProtocol",
          "TargetGroupHealthCheckTimeout",
          "TargetGroupHealthyThreshold",
          "TargetGroupUnhealthyThreshold",
          "TargetGroupValidHTTPCode",
          "TargetGroupDeregistrationDelayTimeout",
          "TargetGroupSlowStartDuration",
          "TargetGroupCookieExpirationPeriod"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1a68ck03fn98r

Classifications:

- Deployment | Advanced stack components | S3 storage | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create S3 bucket",
  "description": "Create an Amazon S3 bucket for cloud storage.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-s2b72beb200000000.",
      "type": "string",
      "enum": [
        "stm-s2b72beb200000000"
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name that is used in the Console.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
```

```
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "BucketName": {
          "description": "A name for the S3 bucket. The S3 bucket name must contain
only lowercase letters, numbers, periods (.), and hyphens (-). The name must be unique
across all existing bucket names in Amazon S3.",
          "type": "string",
          "pattern": "^(?!ams|aws|mc|cf-templates)[a-z0-9]([-.a-z0-9]+)[a-z0-9]$",
          "minLength": 3,
          "maxLength": 63
        },
```

```
        "ServerSideEncryption": {
          "description": "Default encryption for a bucket using server-side encryption
 with either Amazon S3-managed keys (SSE-S3) or AWS KMS-managed keys (SSE-KMS). Use
 None to disable default encryption. Default is KmsManagedKeys.",
          "type": "string",
          "enum": [
            "None",
            "S3ManagedKeys",
            "KmsManagedKeys"
          ]
        },
        "KMSKeyId": {
          "description": "The AWS KMS master key ID used for the ServerSideEncryption
 KMS encryption. Applicable only if ServerSideEncryption = KmsManagedKeys. Leave blank
 to use the default encryption key.",
          "type": "string",
          "pattern": "^arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key\\/[a-f0-9]{8}-[a-f0-9]{4}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key\\/mrk-[a-
z0-9]{32}$|^$"
        },
        "Versioning": {
          "description": "The status of versioning for this S3 bucket, either Enabled
 (versioning of stored objects is enabled) or Suspended (versioning is not enabled).
 Default is Suspended.",
          "type": "string",
          "enum": [
            "Enabled",
            "Suspended"
          ]
        },
        "IAMPrincipalsRequiringReadObjectAccess": {
          "description": "List the Identity and Access Management (IAM), or CloudFront
 Origin Access Identity (OAI), or both, Amazon Resource Names (ARNs) that require
 read access to the S3 bucket. For example, arn:aws:iam::123456789012:role/myrole,
 arn:aws:iam::123456789012:user/myuser and/or arn:aws:iam::cloudfront:user/CloudFront
 Origin Access Identity EH1HDMB1FH2TC.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[/\\w+=,.@-]{1,64}$|
^arn:aws:iam::cloudfront:user\\/CloudFront Origin Access Identity E[A-Z0-9]{11,13}$"
          },
          "minItems": 1,
          "uniqueItems": true
```

```
        },
        "IAMPrincipalsRequiringWriteObjectAccess": {
          "description": "List the IAM ARNs that require write access to the S3 bucket.
 For example, arn:aws:iam::123456789012:role/myrole or arn:aws:iam::123456789012:user/
myuser.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[/\\w+=,.@-]{1,64}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "ServicesRequiringReadObjectAccess": {
          "description": "List of AWS services that require read access to the S3
 bucket; for example, logs.us-east-1.amazonaws.com.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z][a-z0-9.-]+.amazonaws.com$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "ServicesRequiringWriteObjectAccess": {
          "description": "List of AWS services that require write access to the S3
 bucket; for example, logs.us-east-1.amazonaws.com.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z][a-z0-9.-]+.amazonaws.com$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "EnforceSecureTransport": {
          "description": "True to enforce HTTPS for object operations, false to not.",
          "type": "boolean",
          "default": true
        },
        "AccessAllowedIpRanges": {
          "description": "List of source IP ranges allowed to access the S3 bucket.
 Leave blank to not have IP-based restrictions.",
          "type": "array",
```

```
          "items": {
            "type": "string"
          },
          "minItems": 0,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "BucketName",
          "Versioning",
          "ServerSideEncryption",
          "KMSKeyId",
          "EnforceSecureTransport",
          "IAMPrincipalsRequiringReadObjectAccess",
          "IAMPrincipalsRequiringWriteObjectAccess",
          "ServicesRequiringReadObjectAccess",
          "ServicesRequiringWriteObjectAccess",
          "AccessAllowedIpRanges"
        ]
      },
      "required": [
        "BucketName"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
```

```
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1aqsjf86w6vxg

Classifications:

- Deployment | Advanced stack components | EC2 stack | Create (with additional volumes)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EC2 Stack With Additional Volumes",
  "description": "Create an Amazon Elastic Compute Cloud (EC2) instance with up to five
 additional volumes.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
```

```
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-nn8v8ffhcal611bmp.",
      "type": "string",
      "enum": [
        "stm-nn8v8ffhcal611bmp"
      ],
      "default": "stm-nn8v8ffhcal611bmp"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
```

```
   "Parameters": {
     "type": "object",
     "properties": {
       "InstanceAmiId": {
         "type": "string",
         "description": "The AMI to use to create the EC2 instance, in the form
ami-0123abcd or ami-01234567890abcdef.",
         "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$"
       },
       "InstanceCoreCount": {
         "type": "integer",
         "description": "The number of CPU cores for the instance. If you set this,
you need to specify a value for InstanceThreadsPerCore.",
         "minimum": 0,
         "maximum": 224,
         "default": 0
       },
       "InstanceThreadsPerCore": {
         "type": "integer",
         "description": "The number of threads per CPU core. If you set this, you need
to specify a value for InstanceCoreCount.",
         "minimum": 0,
         "maximum": 2,
         "default": 0
       },
       "InstanceDetailedMonitoring": {
         "type": "string",
         "description": "True to turn on detailed monitoring for your instances. False
to turn off detailed monitoring for your instances and set it to basic monitoring.
EC2 detailed monitoring provides more frequent metrics, published at one-minute
intervals, instead of the five-minute intervals used in Amazon EC2 basic monitoring.
Detailed monitoring does incur charges. For more information, see AWS CloudWatch
documentation.",
         "enum": [
           "true",
           "false"
         ]
       },
       "InstanceEBSOptimized": {
         "type": "string",
         "description": "True for the instance to be optimized for Amazon Elastic
Block Store (EBS) I/O, false for it to not be. If you set this to true, choose an
InstanceType that supports EBS optimization.",
         "enum": [
```

```
            "true",
            "false"
          ]
        },
        "InstanceProfile": {
          "type": "string",
          "description": "An IAM instance profile name defined in your account. The
default is customer-mc-ec2-instance-profile.",
          "pattern": "^[a-zA-Z0-9_.=@,+-]{1,128}$"
        },
        "InstanceRootVolumeIops": {
          "type": "integer",
          "description": "The IOPS to use for the root volume, if
InstanceRootVolumeType = io1, io2 or gp3. If InstanceRootVolumeType is not io1, io2 or
gp3, any value provided here is ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "InstanceRootVolumeName": {
          "type": "string",
          "description": "The device name of the root volume for the instance;
for example, /dev/xvda or /dev/sda1. Specify this, and InstanceRootVolumeSize and
InstanceRootVolumeType, to make changes to any or all of these parameters. Leave
blank for the values for those three parameters to be drawn from the InstanceAmiId.
Specifying an InstanceRootVolumeName that does not match that setting in the
InstanceAmiId may result in instance launch failures or making changes to the wrong
volume. Note that setting a value prohibits updating the value with the EC2 instance
stack Update (with additional volumes) ct (ct-1o1x2itfd6rk8) later.",
          "enum": [
            "",
            "/dev/sda1",
            "/dev/xvda"
          ]
        },
        "InstanceRootVolumeSize": {
          "type": "integer",
          "description": "The size, in GiB, of the root volume for the instance.
To change this from the value set in the InstanceAmiId, you must also specify
InstanceRootVolumeName. If no value is provided for InstanceRootVolumeName, any value
provided here is ignored.",
          "minimum": 8,
          "maximum": 16384
        },
        "InstanceRootVolumeType": {
```

```
          "type": "string",
          "description": "The instance type of the root volume for the instance.
 To change this from the value set in the InstanceAmiId, you must also specify
 InstanceRootVolumeName. If no value is provided for InstanceRootVolumeName, any value
 provided here is ignored. Choose io1, io2, gp2 or gp3 for SSD-backed volumes optimized
 for transactional workloads. Choose sc1 or st1 for HDD-backed volumes optimized
 for large streaming workloads. Choose standard for HDD-backed volumes suitable for
 workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3"
          ]
        },
        "RootVolumeKmsKeyId": {
          "description": "The ID, or ARN, of the KMS master key to be used to encrypt
 the root volume. Specify default to use the default EBS KMS Key. Leave blank to not
 encrypt the root volume. Note that, if a value is set, the InstanceRootVolumeName must
 also be specified for KMS encryption settings on the root volume to take effect.",
          "type": "string",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "InstancePrivateStaticIp": {
          "type": "string",
          "description": "The static IP address for the instance."
        },
        "InstanceSecondaryPrivateIpAddressCount": {
          "type": "integer",
          "description": "The number of secondary private IP addresses that EC2
 automatically assigns to the primary network interface. The number of secondary IP
 addresses that can be assigned is dependent on the type of instance used.",
          "minimum": 0
        },
        "InstanceSubnetId": {
          "type": "string",
          "description": "The subnet that you want to launch the instance into, in the
 form subnet-0123abcd or subnet-01234567890abcdef.",
          "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
        },
        "InstanceTerminationProtection": {
          "type": "string",
```

```
        "description": "True to prevent the instance from being terminated through
the API, false to allow it. Default is false. Termination protection must be disabled
with an update (ct-1o1x2itfd6rk8) before deleting the stack or performing an update
where instance replacement is required, otherwise failures occur.",
        "enum": [
          "true",
          "false"
        ]
      },
      "InstanceType": {
        "type": "string",
        "description": "The EC2 instance type. Choose an InstanceType that supports
EBS optimization if InstanceEBSOptimized = true.",
        "default": "t3.large"
      },
      "CreditSpecification": {
        "description": "The credit option for CPU Usage. This is only supported with
t2, t3, and t3a, instance types. If your instance is unlikely to require CPU bursting,
choose standard, but note that, once all the CPU credits for that instance are used
up, it will be throttled. For better burst handling, and to not allow throttling,
choose unlimited, but note that additional charges may apply when additional credits
are used.",
        "type": "string",
        "enum": [
          "unlimited",
          "standard"
        ],
        "default": "unlimited"
      },
      "EnforceIMDSV2": {
        "description": "True for the instance to be launched with IMDSv2 enforced.
Default value is True. If you set this to True, make sure your applications are
compatible with IMDSv2. See EC2/IMDS document for more details.",
        "type": "string",
        "enum": [
          "true",
          "false"
        ],
        "default": "true"
      },
      "InstanceUserData": {
        "type": "string",
        "description": "A newline-delimited string where each line is part of a
script to be run on boot."
```

```
        },
        "Volume1Iops": {
          "type": "integer",
          "description": "The IOPS to use for the Volume1 volume, if Volume1Type =
 io1, io2 or gp3. If Volume1Type is not io1, io2 or gp3, any value provided here is
 ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "Volume1Throughput": {
          "type": "integer",
          "description": "The Throughput to use for the Volume1 volume, if Volume1Type
 = gp3. If Volume1Type is not gp3, any value provided here is ignored. Default is
 125.",
          "minimum": 125,
          "maximum": 1000,
          "default": 125
        },
        "Volume1KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume1. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume1.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume1Name": {
          "type": "string",
          "description": "The device name for Volume1 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume1. Leave blank to skip creation of Volume1.",
          "pattern": "^((/dev/)?sd[f-p][1-6]?|(/dev/)?xvd[f-z])$"
        },
        "Volume1Size": {
          "type": "integer",
          "description": "The size of Volume1 in GiB. Defaults to 1 GiB.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume1Snapshot": {
          "type": "string",
          "description": "The EBS snapshot ID to use to create Volume1.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
```

```
        "Volume1Type": {
          "type": "string",
          "description": "The volume type for Volume1. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1"
          ]
        },
        "Volume2Iops": {
          "type": "integer",
          "description": "The IOPS to use for the Volume2 volume, if Volume2Type =
 io1, io2 or gp3. If Volume2Type is not io1, io2 or gp3, any value provided here is
 ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "Volume2Throughput": {
          "type": "integer",
          "description": "The Throughput to use for the Volume2 volume, if Volume2Type
 = gp3. If Volume2Type is not gp3, any value provided here is ignored. Default is
 125.",
          "minimum": 125,
          "maximum": 1000,
          "default": 125
        },
        "Volume2KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume2. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume2.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume2Name": {
          "type": "string",
```

```
          "description": "The device name for Volume2 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume2. Leave blank to skip creation of Volume2.",
          "pattern": "^((/dev/)?sd[f-p][1-6]?|(/dev/)?xvd[f-z])$"
        },
        "Volume2Size": {
          "type": "integer",
          "description": "The size of Volume2 in GiB. Defaults to 1 GiB",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume2Snapshot": {
          "type": "string",
          "description": "The EBS snapshot ID to use to create Volume2.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume2Type": {
          "type": "string",
          "description": "The volume type for Volume2. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1"
          ]
        },
        "Volume3Iops": {
          "type": "integer",
          "description": "The IOPS to use for the Volume3 volume, if Volume3Type =
 io1, io2 or gp3. If Volume3Type is not io1, io2 or gp3, any value provided here is
 ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "Volume3Throughput": {
          "type": "integer",
```

```
          "description": "The Throughput to use for the Volume3 volume, if Volume3Type
 = gp3. If Volume3Type is not gp3, any value provided here is ignored. Default is
 125.",
          "minimum": 125,
          "maximum": 1000,
          "default": 125
        },
        "Volume3KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume3. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume3.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume3Name": {
          "type": "string",
          "description": "The device name for Volume3 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume3. Leave blank to skip creation of Volume3.",
          "pattern": "^((/dev/)?sd[f-p][1-6]?|(/dev/)?xvd[f-z])$"
        },
        "Volume3Size": {
          "type": "integer",
          "description": "The size of Volume3 in GiB. Defaults to 1 GiB.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume3Snapshot": {
          "type": "string",
          "description": "The EBS snapshot ID to use to create Volume3.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume3Type": {
          "type": "string",
          "description": "The volume type for Volume3. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
```

```
            "gp3",
            "sc1",
            "st1"
          ]
        },
        "Volume4Iops": {
          "type": "integer",
          "description": "The IOPS to use for the Volume4 volume, if Volume4Type =
 io1, io2 or gp3. If Volume4Type is not io1, io2 or gp3, any value provided here is
 ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "Volume4Throughput": {
          "type": "integer",
          "description": "The Throughput to use for the Volume4 volume, if Volume4Type
 = gp3. If Volume3Type is not gp3, any value provided here is ignored. Default is
 125.",
          "minimum": 125,
          "maximum": 1000,
          "default": 125
        },
        "Volume4KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume4. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume4.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume4Name": {
          "type": "string",
          "description": "The device name for Volume4 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume4. Leave blank to skip creation of Volume4.",
          "pattern": "^((/dev/)?sd[f-p][1-6]?|(/dev/)?xvd[f-z])$"
        },
        "Volume4Size": {
          "type": "integer",
          "description": "The size of Volume4 in GiB. Defaults to 1 GiB.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume4Snapshot": {
```

```
          "type": "string",
          "description": "The EBS snapshot ID to use to create Volume4.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume4Type": {
          "type": "string",
          "description": "The volume type for Volume4. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1"
          ]
        },
        "Volume5Iops": {
          "type": "integer",
          "description": "The IOPS to use for the Volume5 volume, if Volume5Type =
 io1, io2 or gp3. If Volume5Type is not io1, io2 or gp3, any value provided here is
 ignored.",
          "minimum": 0,
          "maximum": 64000
        },
        "Volume5Throughput": {
          "type": "integer",
          "description": "The Throughput to use for the Volume5 volume, if Volume5Type
 = gp3. If Volume5Type is not gp3, any value provided here is ignored. Default is
 125.",
          "minimum": 125,
          "maximum": 1000,
          "default": 125
        },
        "Volume5KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume5. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume5.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
```

```
        },
        "Volume5Name": {
          "type": "string",
          "description": "The device name for Volume5 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume5. Leave blank to skip creation of Volume5.",
          "pattern": "^((/dev/)?sd[f-p][1-6]?|(/dev/)?xvd[f-z])$"
        },
        "Volume5Size": {
          "type": "integer",
          "description": "The size of Volume5 in GiB. Defaults to 1 GiB.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume5Snapshot": {
          "type": "string",
          "description": "The EBS snapshot ID to use to create Volume5.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume5Type": {
          "type": "string",
          "description": "The volume type for Volume5. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceAmiId",
          "InstanceSubnetId",
          "InstanceDetailedMonitoring",
          "InstanceEBSOptimized",
          "InstanceProfile",
          "InstanceCoreCount",
```

```
            "InstanceThreadsPerCore",
            "InstanceRootVolumeIops",
            "InstanceRootVolumeName",
            "InstanceRootVolumeSize",
            "InstanceRootVolumeType",
            "RootVolumeKmsKeyId",
            "InstancePrivateStaticIp",
            "InstanceSecondaryPrivateIpAddressCount",
            "InstanceType",
            "CreditSpecification",
            "InstanceUserData",
            "InstanceTerminationProtection",
            "EnforceIMDSV2",
            "Volume1Name",
            "Volume1Size",
            "Volume1Type",
            "Volume1KmsKeyId",
            "Volume1Iops",
            "Volume1Throughput",
            "Volume1Snapshot",
            "Volume2Name",
            "Volume2Size",
            "Volume2Type",
            "Volume2KmsKeyId",
            "Volume2Iops",
            "Volume2Throughput",
            "Volume2Snapshot",
            "Volume3Name",
            "Volume3Size",
            "Volume3Type",
            "Volume3KmsKeyId",
            "Volume3Iops",
            "Volume3Throughput",
            "Volume3Snapshot",
            "Volume4Name",
            "Volume4Size",
            "Volume4Type",
            "Volume4KmsKeyId",
            "Volume4Iops",
            "Volume4Throughput",
            "Volume4Snapshot",
            "Volume5Name",
            "Volume5Size",
            "Volume5Type",
```

```
          "Volume5KmsKeyId",
          "Volume5Iops",
          "Volume5Throughput",
          "Volume5Snapshot"
        ]
      },
      "required": [
        "InstanceAmiId",
        "InstanceSubnetId"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1ax768xtu8c9q

Classifications:

- Management | Advanced stack components | S3 storage | Manage lifecycle configuration

```
{
```

```
 "$schema": "http://json-schema.org/draft-04/schema#",
 "name": "Manage Lifecycle Configuration",
 "description": "Add a new lifecycle configuration, or replace an existing one for an
Amazon S3 bucket.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-PutBucketLifecycleConfiguration.",
     "type": "string",
     "enum": [
       "AWSManagedServices-PutBucketLifecycleConfiguration"
     ],
     "default": "AWSManagedServices-PutBucketLifecycleConfiguration"
   },
   "Region": {
     "description": "The AWS Region in which the AWS resource is located, in the form
us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "BucketName": {
         "description": "The name of the S3 bucket for the lifecycle configuration.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^(?!(mc|ams|awsms)-)[a-z0-9][-.a-z0-9]{1,61}[a-z0-9]$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "LifecycleConfiguration": {
         "description": "The lifecycle configuration in JSON format.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^\\s*\\{\\s*\"Rules\"\\s*:\\s*\\[.*\\]\\s*\\}\\s*$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "ReplaceExisting": {
```

```
          "description": "True to replace the existing lifecycle configuration, False
 to append the new configuration to the existing value. Default is False.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Verification": {
          "description": "A lifecycle policy can be used to delete all objects in a
 bucket. To prevent accidental deletion, please ensure you have entered the correct
 bucket name and the correct lifecycle policy configuration. Enter the value \"confirm
\" in this parameter once you have verified this.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "confirm"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "MinimumNumberOfDaysBeforeExpiration": {
          "description": "The minimum number of days before a rule in the lifecycle
 configuration can expire an object. The value must be greater than one.",
          "type": "array",
          "items": {
            "type": "integer",
            "minimum": 2,
            "maximum": 7300
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
```

```
          "BucketName",
          "LifecycleConfiguration",
          "ReplaceExisting",
          "Verification",
          "MinimumNumberOfDaysBeforeExpiration"
        ]
      },
      "required": [
        "BucketName",
        "LifecycleConfiguration",
        "Verification",
        "MinimumNumberOfDaysBeforeExpiration"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1ay83wy4vxa3k

Classifications:

- Management | AWS Backup | Backup plan | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update AWS Backup Plan",
```

```
 "description": "Update an existing backup plan. Please note that any changes that you
make to a backup plan have no effect on existing backups created by the backup plan.
The changes apply only to backups that are created in the future.",
 "type": "object",
 "properties": {
   "BackupPlanName": {
     "description": "The name of the backup plan to be updated.",
     "type": "string",
     "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
   },
   "ResourceTagKey": {
     "type": "string",
     "description": "The tag key (case sensitive) of the resources to be backed up.
For example, if you want to use a tag key:value pair like 'Department:accounting',
you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
ResourceTagValue.",
     "minLength": 1,
     "maxLength": 127
   },
   "ResourceTagValue": {
     "type": "string",
     "description": "The tag value (case sensitive) of the resources to be backed up.
For example, if you want to use a tag key:value pair like 'Department:accounting',
you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
ResourceTagValue.",
     "minLength": 1,
     "maxLength": 255
   },
   "WindowsVSS": {
     "type": "string",
     "description": "Enabled to use the Windows Volume Shadow Copy Service (VSS)
backup option in AWS Backup. Disabled to create a regular backup. Default is disabled.
If the application has VSS writer registered with Windows VSS, then AWS Backup creates
a snapshot that will be consistent for that application. To learn more, see AWS Backup
documentation \"Creating Windows VSS backups.\"",
     "enum": [
       "disabled",
       "enabled"
     ],
     "default": "disabled"
   },
   "BackupRuleName": {
     "description": "The name of the existing rule in the specified backup plan to be
updated.",
```

```
      "type": "string",
      "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
    },
    "BackupRuleVault": {
      "type": "string",
      "description": "The name of the AWS Backup vault to be used in the AWS Backup
plan rule.",
      "pattern": "^[a-zA-Z0-9\\-\\_]{2,50}$",
      "default": "ams-custom-backups"
    },
    "BackupRuleCompletionWindowMinutes": {
      "type": "integer",
      "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
      "minimum": 1,
      "maximum": 99000
    },
    "BackupRuleScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$"
    },
    "BackupRuleDeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that a backup is deleted, valid
values are between 1 and 35600. If the value is 0 or not specified, the backup never
expires.",
      "minimum": 0,
      "maximum": 35600
    },
    "BackupRuleMoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that a backup is moved to cold
storage, valid values are between 1 and 35600. If the value is 0 or not specified, the
backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600
    },
    "BackupRuleStartWindowMinutes": {
      "type": "integer",
```

```
      "description": "The period of time, in minutes, after a backup is scheduled to
 wait before a job is canceled if it doesn't start successfully.",
      "minimum": 60,
      "maximum": 99000
    },
    "BackupRuleRecoveryPointTagKey": {
      "type": "string",
      "description": "A key for the tag that is assigned to all created recovery points
 for the backup rule.",
      "minLength": 1,
      "maxLength": 127
    },
    "BackupRuleRecoveryPointTagValue": {
      "type": "string",
      "description": "A value for the BackupRuleRecoveryPointTagKey.",
      "minLength": 1,
      "maxLength": 255
    },
    "BackupRuleEnableContinuousBackup": {
      "type": "string",
      "description": "True to create a continuous backup rule, false to not create the
 rule. With continuous backups, you can restore your AWS Backup-supported resource by
 rewinding it back to a specific time that you choose, within 1 second of precision
 (going back a maximum of 35 days). You can do this during the PITR(Point-In-Time
 Recovery) restore process, where the AWS Backup console displays a Restore time
 section.",
      "enum": [
        "true",
        "false"
      ]
    },
    "BackupRuleCopyActionsDestVaultArn": {
      "type": "string",
      "description": "For backup plan rule: The Amazon Resource Name (ARN) of the
 destination backup vault for the copied backup.",
      "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-z]+-
[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
    },
    "BackupRuleCAMoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "For backup plan rule copy actions: The number of days after
 creation before the recovery point is moved to cold storage, valid values are between
 1 and 35600. If the value is 0 or not specified, the backup never moves to cold
 storage. Only Amazon EFS file system backups can be transitioned to cold storage.",
```

```
      "minimum": 0,
      "maximum": 35600
    },
    "BackupRuleCopyActionsDeleteAfterDays": {
      "type": "integer",
      "description": "For backup plan rule copy actions: The number of days after
creation that a recovery point is deleted, valid values are between 1 and 35600. If
the value is 0 or not specified, the backup never expires.",
      "minimum": 0,
      "maximum": 35600
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "BackupPlanName",
      "ResourceTagKey",
      "ResourceTagValue",
      "WindowsVSS",
      "BackupRuleName",
      "BackupRuleVault",
      "BackupRuleCompletionWindowMinutes",
      "BackupRuleScheduleExpression",
      "BackupRuleDeleteAfterDays",
      "BackupRuleMoveToColdStorageAfterDays",
      "BackupRuleStartWindowMinutes",
      "BackupRuleRecoveryPointTagKey",
      "BackupRuleRecoveryPointTagValue",
      "BackupRuleEnableContinuousBackup",
      "BackupRuleCopyActionsDestVaultArn",
      "BackupRuleCAMoveToColdStorageAfterDays",
      "BackupRuleCopyActionsDeleteAfterDays",
      "Priority"
    ]
  },
```

```
  "additionalProperties": false,
  "required": [
    "BackupPlanName",
    "BackupRuleName",
    "BackupRuleVault"
  ]
}
```

Schema for Change Type ct-1b8fudnqq7m8r

Classifications:

- Management | Monitoring and notification | GuardDuty IP set | Delete (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete GuardDuty IPSet",
  "description": "Use to delete an Amazon GuardDuty IPSet instance which is a list of
 trusted IP addresses that have been whitelisted for highly secure communication with
 your AWS environment.",
  "type": "object",
  "properties": {
    "DetectorId": {
      "description": "The detector ID that specifies the GuardDuty service whose IPSet
 you want to delete.",
      "pattern": "^[a-fA-F0-9]{32}$|^$",
      "type": "string"
    },
    "IpSetId": {
      "description": "The unique ID that specifies the IPSet that you want to
 delete.",
      "type": "string",
      "minLength": 1
    },
    "Region": {
      "description": "Region to use in the form of us-east-1.",
      "type": "string",
      "minLength": 1
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
```

```
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "IpSetId",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "IpSetId",
    "Region"
  ]
}
```

Schema for Change Type ct-1c0jrxd3su5oe

Classifications:

- Deployment | Advanced stack components | RDS snapshot | Copy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Copy RDS DB Snapshot",
  "description": "Create a KMS key encrypted copy of an Amazon Relational Database
 Service (Amazon RDS) DB snapshot. If you are copying a snapshot shared from
 another AWS account, it must be located in the same region in which the document is
 executed.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CopyDbSnapshot.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-CopyDbSnapshot"
      ],
      "default": "AWSManagedServices-CopyDbSnapshot"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}|^$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SourceDbSnapshotArn": {
          "description": "The Amazon Resource Name (ARN) of the DB snapshot to be
 copied.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:rds:[a-z0-9-]+:[0-9]{12}:snapshot:[a-zA-Z][a-zA-
Z0-9-:]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TargetDbSnapshotIdentifier": {
          "description": "An identifier for the target DB snapshot.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KmsKeyId": {
          "description": "An AWS Key Management Service (KMS) key to encrypt the DB
 snapshot with. The KMS key is the KMS Key ARN or the KMS key identifier.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$"
          },
          "minItems": 1,
```

```
          "maxItems": 1
        },
        "SourceRegion": {
          "description": "The AWS Region where the source snapshot is located. Leave
blank if the source snapshot is located in the same region in which the document is
executed.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}|^$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "OptionGroupName": {
          "description": "The name of an option group to associate with the copy of the
snapshot. Specify this option if you are copying a snapshot from one AWS Region to
another, and your DB instance uses a nondefault option group. If copying across AWS
Regions, and your source DB instance uses Transparent Data Encryption for Oracle or
Microsoft SQL Server, you must specify this option. For more information, see Option
Group Considerations in the Amazon RDS User Guide.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9-]{0,255}$"
          },
          "minItems": 0,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SourceDbSnapshotArn",
          "TargetDbSnapshotIdentifier",
          "KmsKeyId",
          "SourceRegion",
          "OptionGroupName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SourceDbSnapshotArn",
        "TargetDbSnapshotIdentifier",
        "KmsKeyId"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1d2fml15b9eth

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication task

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS replication task.",
  "description": "Use to create a Database Migration Service (DMS) replication task.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
```

```
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-eos7uq0usnmeggdet",
```

```
      "type": "string",
      "enum": [
        "stm-eos7uq0usnmeggdet"
      ],
      "default": "stm-eos7uq0usnmeggdet"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
 the change. This will not prolong execution, but the RFC fails if the change is not
 completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CdcStartTime": {
          "type": "string",
          "description": "When the DMS starts change data capture (CDC), in epoch time
 (milliseconds). For example, for CDC to start on Thursday August 9, 20018 1:02:49
 AM (UTC), enter 1533776569. Must not be a future time and not all source endpoints
 support CDC start time.",
          "pattern": "^$|^[0-9]*$",
          "default": ""
        },
        "MigrationType": {
          "type": "string",
          "description": "The migration type or method. To migrate existing data use
 full-load, to migrate existing data and replicate ongoing changes use full-load-and-
cdc, to replicate data changes only use cdc.",
          "enum": [
            "full-load",
            "full-load-and-cdc",
            "cdc"
          ]
        },
        "ReplicationInstanceArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the DMS replication
 instance, in the form arn:aws:dms:REGION:ACCOUNTID:rep:ABAICDVER4V47TYTAA3U3SE7YM.",
          "pattern": "^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:rep:[a-zA-Z0-9]+$"
        },
```

```
        "ReplicationTaskIdentifier": {
          "type": "string",
          "description": "An identifier for the task. Use to give the task a name or
label.",
          "pattern": "^$|(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$",
          "default": ""
        },
        "ReplicationTaskSettings": {
          "type": "string",
          "description": "A JSON document defining settings for the task. For example,
task metadata settings, logging settings etc. For large inputs, we recommend removing
extra whitespaces.",
          "default": "",
          "maxLength": 4096
        },
        "SourceEndpointArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of
the DMS source endpoint for the task to use, in the form
arn:aws:dms:REGION:ACCOUNTID:endpoint:ABAICDMTD4V47TYTAA3U3SE7YM.",
          "pattern": "^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:endpoint:[A-Z0-9]+$"
        },
        "TableMappings": {
          "type": "string",
          "description": "A JSON document to set rules for schema mapping, the mapping
method, transformation and filters."
        },
        "TargetEndpointArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of
the DMS target endpoint for the task to use, in the form
arn:aws:dms:REGION:ACCOUNTID:endpoint:XYAICDMTD4V47TYTAA3U3SE7YM.",
          "pattern": "^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:endpoint:[A-Z0-9]+$"
        }
      },
      "metadata": {
        "ui:order": [
          "ReplicationTaskIdentifier",
          "MigrationType",
          "SourceEndpointArn",
          "TargetEndpointArn",
          "ReplicationInstanceArn",
          "TableMappings",
          "ReplicationTaskSettings",
```

```
          "CdcStartTime"
        ]
      },
      "required": [
        "MigrationType",
        "ReplicationInstanceArn",
        "SourceEndpointArn",
        "TableMappings",
        "TargetEndpointArn"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1d55pi44ff21u

Classifications:

- Management | Advanced stack components | DNS (private) | Update

```
{
```

```
 "$schema": "http://json-schema.org/draft-04/schema#",
 "name": "Update Private DNS Record Sets",
 "description": "Update an existing Route 53 DNS Hosted Zone with the supplied
resource record set.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-CreateAddRoute53Resources.",
     "type": "string",
     "enum": [
       "AWSManagedServices-CreateAddRoute53Resources"
     ],
     "default": "AWSManagedServices-CreateAddRoute53Resources"
   },
   "Region": {
     "description": "The AWS Region in which the AWS resource is located, in the form
us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "description": "Specifications for the Stack.",
     "type": "object",
     "properties": {
       "HostedZoneId": {
         "description": "The HostedZoneId that is to be updated. Supply either the
HostedZoneId or the StackId but not both.",
         "type": "string",
         "pattern": "^$|^[a-zA-Z][a-zA-Z0-9]{1,32}$"
       },
       "StackId": {
         "description": "The StackId that is required to be updated. Supply either the
HostedZoneId or the StackId but not both.",
         "type": "string",
         "pattern": "^$|^stack-[a-z0-9]{17}$"
       },
       "RecordSet": {
         "description": "A JSON of resource records for the hosted zone.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^\\s*\\{\\s*\"RecordSet\"\\s*:\\s*\\[.*\\]\\s*\\}\\s*$"
         },
         "minItems": 1,
```

```
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "HostedZoneId",
          "StackId",
          "RecordSet"
        ]
      },
      "required": [
        "RecordSet"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1d84keiri1jhg

Classifications:

- Deployment | Advanced stack components | KMS key | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create KMS key",
  "description": "Request a KMS key with a predefined key policy.",
  "type": "object",
```

```
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name used in the Console.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "minLength": 1,
           "maxLength": 255
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Key",
           "Value"
         ]
       },
```

```
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-enf1j068fhg34vugt",
      "type": "string",
      "enum": [
        "stm-enf1j068fhg34vugt"
      ],
      "default": "stm-enf1j068fhg34vugt"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Alias": {
          "type": "string",
          "description": "An alias for the customer master key (CMK). The alias must
not begin with \"aws/\".",
          "pattern": "^$|(?!aws/)^[a-zA-Z0-9:/_-]+$"
        },
        "EnableKeyRotation": {
          "type": "string",
          "description": "True for automatic rotation of the key material for the
specified CMK, false for no automatic rotation. Default is true.",
          "enum": [
            "true",
            "false"
          ]
        },
```

```
        "Description": {
          "type": "string",
          "description": "A description for the CMK.",
          "maxLength": 8192,
          "minLength": 1
        },
        "PendingWindow": {
          "type": "integer",
          "description": "The number of days in the waiting period before AWS KMS
deletes the CMK. Default is 30.",
          "minimum": 7,
          "maximum": 30
        },
        "IAMPrincipalsRequiringDecryptPermissions": {
          "type": "array",
          "description": "List of IAM ARNs that require permission to
decrypt using the CMK; for example arn:aws:iam::123456789012:role/myrole or
arn:aws:iam::123456789012:user/myuser.",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[\\w+=,.@-]{1,64}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "IAMPrincipalsRequiringEncryptPermissions": {
          "type": "array",
          "description": "List of IAM ARNs that require permission to
encrypt using the CMK; for example arn:aws:iam::123456789012:role/myrole or
arn:aws:iam::123456789012:user/myuser.",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[\\w+=,.@-]{1,64}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "IAMPrincipalsRequiringGrantsPermissions": {
          "type": "array",
          "description": "List of IAM ARNs, or account IDs, allowed to use this CMK for
key grants; for example arn:aws:iam::123456789012:role/myrole or 123456789012.",
          "items": {
            "type": "string",
```

```
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[\\w+=,.@-]{1,64}$|^\
\d{12}$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "LimitGrantsToAWSResources": {
          "type": "string",
          "description": "True to allow only AWS services that are integrated with AWS
 KMS to perform the grant operation on the user's behalf, false to allow any principal
 provided in IAMPrincipalsRequiringGrantsPermissions. Default is false.",
          "enum": [
            "true",
            "false"
          ]
        },
        "EnforceEncryptionContextKeys": {
          "type": "string",
          "description": "True to enforce use of encryption context keys in
 cryptographic operations, false to not. To define the encryption context keys, use
 AllowedEncryptionContextKeys. Default is false.",
          "enum": [
            "true",
            "false"
          ]
        },
        "AllowedEncryptionContextKeys": {
          "type": "array",
          "description": "List of encryption context keys that must be present in
 requests for cryptographic operations. If supplied, all cryptographic operations must
 have one of the context keys from this list.",
          "items": {
            "type": "string"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "AllowServiceRolesAccessKMSKeys": {
          "type": "array",
          "description": "Provide KMS key access to AWS services, by providing the
 endpoint in the form, ec2.us-east-1.amazonaws.com. Then the specified AWS service
 can use the CMK with limited permissions (list and create grants; describe, encrypt,
 decrypt, and reencrypt key; and generate data key).",
          "items": {
```

```
            "type": "string",
            "pattern": "^([a-zA-Z0-9-.]+\\.)+amazonaws\\.com$"
          },
          "minItems": 1,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "Alias",
          "Description",
          "EnableKeyRotation",
          "PendingWindow",
          "IAMPrincipalsRequiringDecryptPermissions",
          "IAMPrincipalsRequiringEncryptPermissions",
          "IAMPrincipalsRequiringGrantsPermissions",
          "LimitGrantsToAWSResources",
          "EnforceEncryptionContextKeys",
          "AllowedEncryptionContextKeys",
          "AllowServiceRolesAccessKMSKeys"
        ]
      },
      "required": [
        "Description"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
```

```
    "StackTemplateId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1dmlg9g1l91h6

Classifications:

- Management | Access | Stack admin access | Grant

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Grant Stack Admin access",
  "description": "Request admin access for one or more users for one or more stacks.
 The maximum access time is 12 hours.",
  "type": "object",
  "properties": {
    "DomainFQDN": {
      "description": "The FQDN for the user accounts to grant access to.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "StackIds": {
      "description": "A minimum of one stack ID is required.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^stack-[a-z0-9]{17}$|^SC-[0-9]{12}-pp-[a-zA-Z0-9]{13}$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "TimeRequestedInHours": {
      "description": "The amount of time, in hours, requested for access to the
 instance. Access is terminated after this time.",
      "type": "integer",
      "minimum": 1,
      "default": 1
    },
```

```
    "Usernames": {
      "description": "One or more Active Directory user names used to grant access.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "VpcId": {
      "description": "The ID of the VPC that contains the stacks where access is
 required, in the form of vpc-12345678 or vpc-1234567890abcdef0.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackIds",
      "Usernames",
      "DomainFQDN",
      "TimeRequestedInHours"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DomainFQDN",
    "StackIds",
    "Usernames",
    "VpcId"
  ]
}
```

Schema for Change Type ct-1e0xmuy1diafq

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Update entity or policy (read-write permissions)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Entity or Policy (read-write permissions)",
  "description": "Update Identity and Access Management (IAM) role or policy with read-
write permissions. You must have enabled this feature with change type ct-1706xvvk6j9hf
 before submitting this request. Automated IAM provisioning with read-write permissions
 runs over 200 validations to help ensure successful outcomes.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin"
      ],
      "default": "AWSManagedServices-HandleAutomatedIAMProvisioningUpdate-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
```

```
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ValidateOnly": {
          "description": "Yes to validate the IAM role or policy updated with the
specified parameter values, without updating the entity or policy; No to validate
and update the entity or policy. The validation result is provided as a JSON in the
execution output. In order to implement after validation, create a copy of the RFC and
set the ValidateOnly parameter to No, then submit.",
          "type": "string",
          "default": "No",
          "enum": [
            "Yes",
            "No"
          ]
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "ValidateOnly"
        ]
      },
      "required": [
        "ValidateOnly"
      ]
    },
    "RoleDetails": {
      "type": "object",
      "properties": {
        "Roles": {
          "description": "Update a role.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "RoleName": {
```

```
                "description": "A name of the IAM role to update. The name can be up
 to 64 characters in length, and is limited to characters a-z, A-Z, 0-9, hyphen and
 underscore",
                "type": "string",
                "pattern": "^[a-zA-Z0-9_-]{1,64}$"
              },
              "Description": {
                "description": "A meaningful description for the role.",
                "type": "string",
                "minLength": 0,
                "maxLength": 5200,
                "default": ""
              },
              "AssumeRolePolicyDocument": {
                "description": "A JSON policy document, defining which entities can
 assume the role, you are updating the current policy document associated to the
 role with. Paste the contents into the input. Content provided replaces existing
 content.",
                "type": "string",
                "minLength": 2,
                "maxLength": 131072
              },
              "ManagedPolicyArns": {
                "description": "A list of Amazon Resource Names (ARNs) of the IAM
 managed policies that you want to attach to the role. Both AWS managed policies and
 customer managed policies are allowed. You must include the list of managed policy
 ARNs currently attached to the role that you wish to keep attached. Value provided
 replaces existing list of ARNs attached to the role.",
                "type": "array",
                "items": {
                  "type": "string",
                  "pattern": "^arn:[\\w+=/,.@-]+:iam::[0-9]{12}:policy(/[\\w+=/,.@-]+)?
$|^arn:[\\w+=/,.@-]+:iam::aws:policy(/[\\w+=/,.@-]+)?$"
                },
                "minItems": 0,
                "maxItems": 20
              },
              "MaxSessionDuration": {
                "description": "The maximum session duration (in seconds) that you want
 to set for the specified role. If you do not specify a value for this setting, the
 default value of one hour is applied. This setting can have a value from 1 hour to 4
 hours. The MaxSessionDuration time begins with the assumption of the role.",
                "type": "string",
                "default": "3600",
```

```
                "pattern": "^(360\\d|36[1-9]\\d|3[7-9]\\d{2}|[4-9]\\d{3}|1[0-3]\\d{3}|
14[0-3]\\d{2}|14400)$"
              },
              "PermissionsBoundary": {
                "description": "The ARN of the policy used to set as the permissions
 boundary for the role. A permissions boundary uses a managed policy to set the maximum
 permissions that an identity-based policy can grant to an IAM entity. ARN provided
 replaces current permission boundary ARN set in the role.",
                "type": "string",
                "default": "",
                "pattern": "^$|^arn:[\\w+=/,.@-]+:iam::[0-9]{12}:policy(/[\\w
+=/,.@-]+)?$"
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "RoleName",
                "Description",
                "AssumeRolePolicyDocument",
                "ManagedPolicyArns",
                "MaxSessionDuration",
                "PermissionsBoundary"
              ]
            },
            "required": [
              "RoleName"
            ]
          },
          "minItems": 0,
          "maxItems": 1,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Roles"
        ]
      }
    },
    "ManagedPolicyDetails": {
      "type": "object",
      "properties": {
```

```
        "Policies": {
          "description": "Update a customer managed policy.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "ManagedPolicyName": {
                "description": "The name of the IAM policy to update. The name can be
up to 128 characters in length, and is limited to characters a-z, A-Z, 0-9, hyphen and
underscore",
                "type": "string",
                "pattern": "^[a-zA-Z0-9_-]{1,128}$"
              },
              "PolicyDocument": {
                "description": "The JSON policy document that you want to use as the
content for the new policy. Paste the content into the input field. Content provided
replaces existing content in the policy.",
                "type": "string",
                "minLength": 2,
                "maxLength": 131072
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "ManagedPolicyName",
                "PolicyDocument"
              ]
            },
            "required": [
              "ManagedPolicyName"
            ]
          },
          "minItems": 0,
          "maxItems": 1,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Policies"
        ]
      }
```

```
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters",
      "RoleDetails",
      "ManagedPolicyDetails"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1e1xtak34nx76

Classifications:

- Management | Other | Other | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create other",
  "description": "Use to request manual creation of a resource.",
  "type": "object",
  "properties": {
    "Comment": {
      "description": "The description of the change.",
      "type": "string",
      "maxLength": 5000
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
```

```
        "Medium",
        "High"
      ]
    },
    "RelatedIds": {
      "description": "(Optional) IDs of resources related to the change request.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 1000,
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "required": [
    "Comment"
  ],
  "metadata": {
    "ui:order": [
      "Comment",
      "RelatedIds",
      "Priority"
    ]
  }
}
```

Schema for Change Type ct-1eft8s6vdhz0w

Classifications:

- Management | Directory Service | DNS | Update record permission

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update DNS Record Permission",
  "description": "Grant permissions to the computer object to update DNS records after
 failover. For multi-account landing zone (MALZ), use this change type in the shared
 services account.",
  "type": "object",
  "properties": {
```

```
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateDNSRecordsPermission-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateDNSRecordsPermission-Admin"
      ],
      "default": "AWSManagedServices-UpdateDNSRecordsPermission-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RecordNames": {
          "description": "A list of comma separated DNS record names.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[A-Za-z0-9-_,]{1,1000}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "RecordNames"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RecordNames"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
```

```
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1eiczxw8ihc18

Classifications:

- Management | Advanced stack components | AMI | Share

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Share AMI",
  "description": "Use to share an AMI with another AMS account.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "TargetAwsAccountId": {
      "pattern": "^[0-9]{12}$",
      "description": "ID of the AWS account the AMI will be shared with, in the form
 123456789012. The account must already be onboarded to AMS.",
      "type": "string"
    },
    "AmiId": {
      "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$",
      "description": "ID of the AMI to share, in the form ami-12345678 or
 ami-123456789012345ab.",
      "type": "string"
    }
  },
  "required": [
    "AmiId",
    "TargetAwsAccountId"
  ]
}
```

Schema for Change Type ct-1erytvmumckoa

Classifications:

- Management | Advanced stack components | Tag | Delete (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Resource Tags (Review Required)",
  "description": "Delete tags from existing, supported resources except those in
 AMS infrastructure stacks (stacks named mc-*). For Autoscaling, EC2, Elastic Load
 Balancing, RDS resources and S3 buckets, use automated CT ct-2zebb2czoxpjd.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the tag operation.",
      "type": "string",
      "maxLength": 5000
    },
    "Resources": {
      "description": "Parameters for up to fifty resources that you want to remove tags
 from.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "ResourceArn": {
            "description": "The ARN or the resource ID of the resource to be tagged.
 Resource ID is allowed only for these resource types: EC2 instance, EBS volume, EBS
 snapshot, AMI, and security group. All other resource types must be provided with the
 full ARN.",
            "type": "string",
            "pattern": "^arn:aws:(|[a-z][a-z0-9-]+):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-
z]+-\\d{1}):(|[0-9]{12}):([^,\\s]+)$|^(ami|i|vol|sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "RemoveTags": {
            "description": "Up to fifty tag keys to remove from the resource.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^(?![aA][mMwW][sS]:)[a-zA-Z0-9\\s_.:/=+\\\\\\-@\\]*]+$",
              "minLength": 1,
```

```
              "maxLength": 127
            },
            "minItems": 1,
            "maxItems": 50,
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "ResourceArn",
            "RemoveTags"
          ]
        },
        "required": [
          "ResourceArn",
          "RemoveTags"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Description",
      "Resources",
      "Priority"
    ]
  },
  "required": [
    "Description",
```

```
    "Resources"
  ]
}
```

Schema for Change Type ct-1ezarc5xph3tq

Classifications:

- Management | Advanced stack components | RDS database stack | Rotate DB certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Rotate RDS DB Certificate",
  "description": "Rotate the DB certificate on an Amazon Relational Database Service
 (RDS) database (DB) instance. Update any client applications that use SSL/TLS and the
 server certificate to connect, to use the new CA certificate beforehand. Not doing
 this will cause an interruption of connectivity between your applications and your
 database.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RotateDbCertificate.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RotateDbCertificate"
      ],
      "default": "AWSManagedServices-RotateDbCertificate"
    },
    "Region": {
      "description": "The AWS Region in which the RDS DB is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBInstanceIdentifier": {
          "description": "RDS DB instance identifier, in the form dbinstance-1.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "(?=[a-zA-Z0-9-]{1,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CertificateIdentifier": {
          "description": "Choose from rds-ca-rsa2048-g1, rds-ca-rsa4096-g1, or rds-ca-
ecc384-g1 to rotate with the latest certificate. Make sure that the certificate applies
 to the database engine. If you have issues with your client-side trust store after
 updating to the latest certificate, then re-submit this RFC and choose rds-ca-2019 to
 revert. After you correct your client-side trust store with the new CA certificate,
 update to the desired certificate again. Note that this workaround is only available
 until August 22, 2024, when the rds-ca-2019 certificate expires.",
          "type": "array",
          "items": {
            "enum": [
              "rds-ca-2019",
              "rds-ca-rsa2048-g1",
              "rds-ca-rsa4096-g1",
              "rds-ca-ecc384-g1"
            ],
            "type": "string",
            "default": "rds-ca-2019"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ApplyImmediately": {
          "description": "True to apply the certificate change immediately. False to
 schedule the change for the next maintenance window. Note that choosing True causes
 the instance to reboot. If applicable, make sure that you have updated your client-
side trust store beforehand.",
          "type": "array",
          "items": {
            "enum": [
              "True",
              "False"
            ],
            "type": "string",
            "default": "False"
          },
          "minItems": 1,
          "maxItems": 1
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "DBInstanceIdentifier",
          "CertificateIdentifier",
          "ApplyImmediately"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBInstanceIdentifier",
        "CertificateIdentifier",
        "ApplyImmediately"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1f9hi4bephqa9

Classifications:

- Management | Managed landing zone | Networking account | Enable TGW propagation

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable TGW Propagation",
```

```
  "description": "Enable the Transit Gateway (TGW) attachment to propagate routes to
 the TGW route table. For multi-account landing zone (MALZ), use this change type in
 the Network account only.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-EnableTGWRouteTablePropagation.",
      "type": "string",
      "enum": [
        "AWSManagedServices-EnableTGWRouteTablePropagation"
      ],
      "default": "AWSManagedServices-EnableTGWRouteTablePropagation"
    },
    "Region": {
      "description": "The AWS Region where the TGW attachment and TGW route table are
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "TransitGatewayAttachmentId": {
          "description": "The TGW attachment ID, in the form tgw-
attach-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-attach-[a-z0-9]{17}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "TransitGatewayRouteTableId": {
          "description": "The TGW route table ID, in the form tgw-
rtb-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-rtb-[a-z0-9]{17}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "TransitGatewayAttachmentId",
          "TransitGatewayRouteTableId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayAttachmentId",
        "TransitGatewayRouteTableId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1fzddqrr20c2i

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Update MaxSessionDuration

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update MaxSessionDuration",
```

```
  "description": "Update the MaxSessionDuration property of an AWS Identity and Access
 Management (IAM) role. This setting determines the maximum duration that can be
 requested using the DurationSeconds parameter when assuming an IAM role.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateIAMRoleMaxSessionDuration.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateIAMRoleMaxSessionDuration"
      ],
      "default": "AWSManagedServices-UpdateIAMRoleMaxSessionDuration"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RoleName": {
          "description": "The name of the IAM role to modify.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(aws-ams-|aws-sentinel-|ams_ssm_|customer_ssm_))[\\w
+=,.@-]+"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "MaxSessionDuration": {
          "description": "The new maximum session duration (in seconds) to set for the
 role. The duration can range from 3600 seconds to 14400 seconds.",
          "type": "array",
          "items": {
            "type": "integer",
            "minimum": 3600,
            "maximum": 14400
          },
          "minItems": 1,
          "maxItems": 1
```

```
        }
      },
      "metadata": {
        "ui:order": [
          "RoleName",
          "MaxSessionDuration"
        ]
      },
      "required": [
        "RoleName",
        "MaxSessionDuration"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1g6x4ev0hnvfn

Classifications:

- Management | AMS Resource Scheduler | Period | Describe

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Describe Resource Scheduler Periods",
  "description": "Describe existing periods used in AMS Resource Scheduler.",
  "type": "object",
  "properties": {
```

```
    "DocumentName": {
      "description": "Must be AWSManagedServices-DescribeScheduleOrPeriods.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DescribeScheduleOrPeriods"
      ],
      "default": "AWSManagedServices-DescribeScheduleOrPeriods"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ConfigurationType": {
          "description": "Specify the value: periods. This explicitly requests that the
Resource Scheduler existing periods be described. The option cannot be left blank; it
must be periods.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "periods"
            ],
            "default": "periods"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ConfigurationType"
        ]
      },
      "required": [
        "ConfigurationType"
      ],
      "additionalProperties": false
    }
  },
```

```
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1gi93jhvj28eg

Classifications:

- Management | Advanced stack components | S3 storage | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update S3 Bucket",
  "description": "Modify the properties of an S3 bucket created using change type ID
 ct-1a68ck03fn98r, version 4.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef. This identifies the AWS Region where the S3 bucket is.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the S3 bucket that you are updating, in the form
 stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the S3 bucket.",
```

```
      "type": "object",
      "properties": {
        "ServerSideEncryption": {
          "description": "Default encryption for an S3 bucket using server-side
 encryption with either Amazon S3-managed keys (SSE-S3) or AWS KMS-managed keys (SSE-
KMS). Use None to disable default encryption.",
          "type": "string",
          "enum": [
            "None",
            "S3ManagedKeys",
            "KmsManagedKeys"
          ]
        },
        "KMSKeyId": {
          "description": "The AWS KMS master key ID used for the ServerSideEncryption
 KMS encryption. Applicable only if ServerSideEncryption = KmsManagedKeys.",
          "type": "string",
          "pattern": "^arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key\\/[a-f0-9]{8}-[a-f0-9]{4}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key\\/mrk-[a-
z0-9]{32}$|^$"
        },
        "Versioning": {
          "description": "The status of versioning for this S3 bucket, either Enabled
 (versioning of stored objects is enabled) or Suspended (versioning is not enabled).",
          "type": "string",
          "enum": [
            "Enabled",
            "Suspended"
          ]
        },
        "IAMPrincipalsRequiringReadObjectAccess": {
          "description": "List the Identity and Access Management (IAM), or CloudFront
 Origin Access Identity (OAI), or both, Amazon Resource Names (ARNs) that require
 read access to the S3 bucket. For example, arn:aws:iam::123456789012:role/myrole,
 arn:aws:iam::123456789012:user/myuser and/or arn:aws:iam::cloudfront:user/CloudFront
 Origin Access Identity EH1HDMB1FH2TC. The list of ARNs provided here replaces the
 existing list in the policy, it does not append to the existing list. To remove all
 ARNs during an update specify None.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[/\\w+=,.@-]{1,64}$|
^arn:aws:iam::cloudfront:user\\/CloudFront Origin Access Identity E[A-Z0-9]{11,13}$|
^None$"
```

```
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "IAMPrincipalsRequiringWriteObjectAccess": {
          "description": "List the IAM ARNs that require write access to the S3 bucket.
 For example, arn:aws:iam::123456789012:role/myrole or arn:aws:iam::123456789012:user/
myuser. The list of ARNs provided here replaces the existing list in the policy, it
 does not append to the existing list. To remove all ARNs during an update, specify
 None.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:iam::\\d{12}:(role|user)\\/[/\\w+=,.@-]{1,64}$|^None
$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "ServicesRequiringReadObjectAccess": {
          "description": "List of AWS services that require read access to the S3
 bucket; for example, logs.us-east-1.amazonaws.com. The list of services provided here
 replaces the existing list in the policy, it does not append to the existing list. To
 remove all AWS services during an update, specify None.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z][a-z0-9.-]+.amazonaws.com$|^None$"
          },
          "minItems": 1,
          "uniqueItems": true
        },
        "ServicesRequiringWriteObjectAccess": {
          "description": "List of AWS services that require write access to the S3
 bucket; for example, logs.us-east-1.amazonaws.com. The list of services provided here
 replaces the existing list in the policy, it does not append to the existing list. To
 remove all AWS services during an update, specify None.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z][a-z0-9.-]+.amazonaws.com$|^None$"
          },
          "minItems": 1,
          "uniqueItems": true
```

```
        },
        "EnforceSecureTransport": {
          "description": "True to enforce HTTPS for object operations. If false, both
HTTP and HTTPS traffic is allowed.",
          "type": "boolean"
        },
        "AccessAllowedIpRanges": {
          "description": "List of source IP ranges allowed to access the S3 bucket.
Leave blank to not have IP-based restrictions. The list of IP ranges provided here
replaces the existing list in the policy, it does not append to the existing list. To
remove all source IP ranges during an update, specify None.",
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 1,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Versioning",
          "ServerSideEncryption",
          "KMSKeyId",
          "EnforceSecureTransport",
          "IAMPrincipalsRequiringReadObjectAccess",
          "IAMPrincipalsRequiringWriteObjectAccess",
          "ServicesRequiringReadObjectAccess",
          "ServicesRequiringWriteObjectAccess",
          "AccessAllowedIpRanges"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
```

```
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1h1tuxn2oxrtf

Classifications:

- Deployment | Advanced stack components | DynamoDB | Create from backup

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DynamoDB From Backup",
  "description": "Create an Amazon DynamoDB stack from backup.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartRestoreJobDynamoDB.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartRestoreJobDynamoDB"
      ],
      "default": "AWSManagedServices-StartRestoreJobDynamoDB"
    },
    "Region": {
      "description": "The AWS Region in which the DynamoDB table is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BackupVaultName": {
          "description": "The name of a logical container where backups are stored.
 The backup vault name is case sensitive and must contain from 2 to 50 alphanumeric
 characters or hyphens.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "maxItems": 1
        },
        "RecoveryPointArn": {
          "description": "The Amazon Resource Name (ARN) that uniquely identifies the
 recovery point to restore.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        },
        "TargetTableName": {
          "description": "The name of the new table to which the backup must be
 restored. The target table name is case sensitive and must contain from 3 to 255
 alphanumeric characters, hyphens, underscores or dots.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\_\\-\\.]{3,255}$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "BackupVaultName",
          "RecoveryPointArn",
          "TargetTableName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BackupVaultName",
        "RecoveryPointArn",
        "TargetTableName"
      ]
    }
  },
  "metadata": {
    "ui:order": [
```

```
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1h5xgl9cr4bzy

Classifications:

- Management | Standard stacks | Stack | Start

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start stack",
  "description": "Use to start all stopped EC2 instances in the specified stack.",
  "type": "object",
  "properties": {
    "StackId": {
      "description": "ID of the stack to start, in the form stack-a1b2c3d4e5f67890e.
 All stopped EC2 instances in the stack will be started.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    }
  },
  "additionalProperties": false,
  "required": [
    "StackId"
  ]
}
```

Schema for Change Type ct-1hzofpphabs3i

Classifications:

- Management | Advanced stack components | DNS (public) | Update


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Public DNS Record Sets",
  "description": "Update an existing Route 53 DNS Hosted Zone with the supplied
 resource record set.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateAddRoute53Resources.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateAddRoute53Resources"
      ],
      "default": "AWSManagedServices-CreateAddRoute53Resources"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "description": "Specifications for the Stack.",
      "type": "object",
      "properties": {
        "HostedZoneId": {
          "description": "The HostedZoneId that is to be updated. Supply either the
 HostedZoneId or the StackId but not both.",
          "type": "string",
          "pattern": "^$|^[a-zA-Z][a-zA-Z0-9]{1,32}$"
        },
        "StackId": {
          "description": "The StackId that is required to be updated. Supply either the
 HostedZoneId or the StackId but not both.",
          "type": "string",
          "pattern": "^$|^stack-[a-z0-9]{17}$"
```

```
        },
        "RecordSet": {
          "description": "A JSON of resource records for the hosted zone.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\s*\\{\\s*\"RecordSet\"\\s*:\\s*\\[.*\\]\\s*\\}\\s*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "HostedZoneId",
          "StackId",
          "RecordSet"
        ]
      },
      "required": [
        "RecordSet"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1i20abktsm05v

Classifications:

- Management | Directory Service | Users and groups | Add group to group


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add AD Group To AD Group",
  "description": "Add an Active Directory (AD) group in the trusted domain to an AD
 group in the AMS managed AD. For multi-account landing zone (MALZ), use this change
 type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddADGroupToADGroup-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddADGroupToADGroup-Admin"
      ],
      "default": "AWSManagedServices-AddADGroupToADGroup-Admin"
    },
    "Region": {
      "description": "The AWS Region where the AMS managed AD is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "NestedGroupName": {
          "description": "The name of the group in the trusted AD to be added to a
 group in the AMS managed AD.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|
\\\\][^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|\\\\]{0,61}[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\
\|]$"
          },
          "maxItems": 1,
```

```
          "minItems": 1
        },
        "GroupName": {
          "description": "The name of the AD group that the nested group is added to.
 The group must exist in AMS managed AD and must belong to the CustomerGroups OU. The
 group scope must be DomainLocal.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|
\\\\][^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|\\\\]{0,61}[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\
\|]$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "TrustedDomainFQDN": {
          "description": "The fully qualified domain name (FQDN) of your domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?![aA][0-9]{12}.[aA][mM][aA][zZ][oO][nN][aA][wW][sS].[cC][oO]
[mM])^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "NestedGroupName",
          "GroupName",
          "TrustedDomainFQDN"
        ]
      },
      "required": [
        "NestedGroupName",
        "GroupName",
        "TrustedDomainFQDN"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1icghmq38rnsn

Classifications:

- Management | Directory Service | DNS | Delete conditional forwarder

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete AD DNS Conditional Forwarder",
  "description": "Delete AD DNS conditional forwarder for a remote domain. For multi-
account landing zone (MALZ), use this change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteADDNSConditionalForwarder-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteADDNSConditionalForwarder-Admin"
      ],
      "default": "AWSManagedServices-DeleteADDNSConditionalForwarder-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
```

```
    "Parameters": {
      "type": "object",
      "properties": {
        "RemoteDomainName": {
          "description": "The fully qualified domain name (FQDN) of the remote
 domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+[.]?$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "RemoteDomainName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RemoteDomainName"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1icrtx8ydvdwe

Classifications:

- Management | Directory Service | DNS | Remove record


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove DNS Record",
  "description": "Remove the specified DNS resource record name, either an A or CNAME,
 or pointer record (PTR), from the specified DNS zone. By default, only the static
 record is removed per specified RecordName for A or CNAME records. Use the RecordData
 parameter to remove duplicates if there are multiple records with the same Host Name
 (RecordType A), either dynamic or static. For a PTR record type, all the static and
 dynamic records will be removed. For multi-account landing zone (MALZ), use this
 change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-RemoveDNSRecord-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveDNSRecord-Admin"
      ],
      "default": "AWSManagedServices-RemoveDNSRecord-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RecordName": {
          "description": "The name of the DNS record (A or CNAME). If it is a pointer
 record (PTR), provide the IPv4 address.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^[a-zA-Z0-9\\-\\_\\-]{1,63}$|^(([0-9]|[1-9][0-9]|1[0-9]{2}|
2[0-4][0-9]|25[0-5])\\.){3}([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RecordType": {
          "description": "The resource record type (A, CNAME, or PTR).",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "A",
              "CNAME",
              "PTR"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RecordData": {
          "description": "The IPv4 address. Use this parameter when there are multiple
 records with the same hostname.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}
([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$|^$",
            "default": ""
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "RecordName",
          "RecordType",
          "RecordData"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RecordName",
```

```
        "RecordType"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1j3503fres5a5

Classifications:

- Deployment | Managed landing zone | Application account | Create VPC

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Application Account VPC",
  "description": "Create a VPC with up to 10 private subnets and up to 5 optional
 public subnets per availability zone (AZ) for two or three AZ's.",
  "type": "object",
  "properties": {
    "VpcName": {
      "description": "A meaningful name for the VPC. Must be unique within this
 application account.",
      "type": "string"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "NumberOfAZs": {
```

```
          "description": "The number of availability zones (AZs) that the VPC supports.
 Options are 2 or 3.",
          "type": "number",
          "minimum": 2,
          "maximum": 3
        },
        "VPCCIDR": {
          "description": "The Classless Inter-Domain Routing (CIDR) for the VPC.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "RouteType": {
          "description": "The AWS Transit Gateway application route table connection
 type. For this VPC to accept connections from other VPCs, use routable. For it to not
 accept those connections, use isolated. The default is routable.",
          "type": "string",
          "enum": [
            "isolated",
            "routable"
          ],
          "default": "routable"
        },
        "TransitGatewayApplicationRouteTableName": {
          "description": "The existing AWS Transit Gateway route table for this
 application account VPC. The default is defaultAppRouteDomain. To create a new
 application route table, use the Create Application Route Table change type.",
          "type": "string",
          "default": "defaultAppRouteDomain"
        },
        "PublicSubnetAZ1CIDR": {
          "description": "The CIDR for the optional first public subnet in availability
 zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnetAZ2CIDR": {
          "description": "The CIDR for the optional first public subnet in availability
 zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
```

```
        "PublicSubnetAZ3CIDR": {
          "description": "The CIDR for the optional first public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet2AZ1CIDR": {
          "description": "The CIDR for the optional second public subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet2AZ2CIDR": {
          "description": "The CIDR for the optional second public subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet2AZ3CIDR": {
          "description": "The CIDR for the optional second public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet3AZ1CIDR": {
          "description": "The CIDR for the optional third public subnet in availability
 zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet3AZ2CIDR": {
          "description": "The CIDR for the optional third public subnet in availability
 zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet3AZ3CIDR": {
```

```
          "description": "The CIDR for the optional third public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet4AZ1CIDR": {
          "description": "The CIDR for the optional fourth public subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet4AZ2CIDR": {
          "description": "The CIDR for the optional fourth public subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet4AZ3CIDR": {
          "description": "The CIDR for the optional fourth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet5AZ1CIDR": {
          "description": "The CIDR for the optional fifth public subnet in availability
 zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet5AZ2CIDR": {
          "description": "The CIDR for the optional fifth public subnet in availability
 zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnet5AZ3CIDR": {
          "description": "The CIDR for the optional fifth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
```

```
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet1AZ1CIDR": {
          "description": "The CIDR for the first private subnet in availability zone
 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet1AZ2CIDR": {
          "description": "The CIDR for the first private subnet in availability zone
 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet1AZ3CIDR": {
          "description": "The CIDR for the first private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet2AZ1CIDR": {
          "description": "The CIDR for the optional second private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet2AZ2CIDR": {
          "description": "The CIDR for the optional second private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet2AZ3CIDR": {
          "description": "The CIDR for the optional second private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
```

```
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet3AZ1CIDR": {
          "description": "The CIDR for the optional third private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet3AZ2CIDR": {
          "description": "The CIDR for the optional third private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet3AZ3CIDR": {
          "description": "The CIDR for the optional third private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet4AZ1CIDR": {
          "description": "The CIDR for the optional fourth private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet4AZ2CIDR": {
          "description": "The CIDR for the optional fourth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet4AZ3CIDR": {
          "description": "The CIDR for the optional fourth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
```

```
        },
        "PrivateSubnet5AZ1CIDR": {
          "description": "The CIDR for the optional fifth private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet5AZ2CIDR": {
          "description": "The CIDR for the optional fifth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet5AZ3CIDR": {
          "description": "The CIDR for the optional fifth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet6AZ1CIDR": {
          "description": "The CIDR for the optional sixth private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet6AZ2CIDR": {
          "description": "The CIDR for the optional sixth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet6AZ3CIDR": {
          "description": "The CIDR for the optional sixth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet7AZ1CIDR": {
```

```
          "description": "The CIDR for the optional seventh private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet7AZ2CIDR": {
          "description": "The CIDR for the optional seventh private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet7AZ3CIDR": {
          "description": "The CIDR for the optional seventh private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet8AZ1CIDR": {
          "description": "The CIDR for the optional eighth private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet8AZ2CIDR": {
          "description": "The CIDR for the optional eighth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet8AZ3CIDR": {
          "description": "The CIDR for the optional eighth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet9AZ1CIDR": {
          "description": "The CIDR for the optional ninth private subnet in
 availability zone 1.",
```

```
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet9AZ2CIDR": {
          "description": "The CIDR for the optional ninth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet9AZ3CIDR": {
          "description": "The CIDR for the optional ninth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet10AZ1CIDR": {
          "description": "The CIDR for the optional tenth private subnet in
 availability zone 1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet10AZ2CIDR": {
          "description": "The CIDR for the optional tenth private subnet in
 availability zone 2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnet10AZ3CIDR": {
          "description": "The CIDR for the optional tenth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        }
      },
      "metadata": {
        "ui:order": [
          "VPCCIDR",
          "NumberOfAZs",
```

```
            "RouteType",
            "TransitGatewayApplicationRouteTableName",
            "PublicSubnetAZ1CIDR",
            "PublicSubnetAZ2CIDR",
            "PublicSubnetAZ3CIDR",
            "PublicSubnet2AZ1CIDR",
            "PublicSubnet2AZ2CIDR",
            "PublicSubnet2AZ3CIDR",
            "PublicSubnet3AZ1CIDR",
            "PublicSubnet3AZ2CIDR",
            "PublicSubnet3AZ3CIDR",
            "PublicSubnet4AZ1CIDR",
            "PublicSubnet4AZ2CIDR",
            "PublicSubnet4AZ3CIDR",
            "PublicSubnet5AZ1CIDR",
            "PublicSubnet5AZ2CIDR",
            "PublicSubnet5AZ3CIDR",
            "PrivateSubnet1AZ1CIDR",
            "PrivateSubnet1AZ2CIDR",
            "PrivateSubnet1AZ3CIDR",
            "PrivateSubnet2AZ1CIDR",
            "PrivateSubnet2AZ2CIDR",
            "PrivateSubnet2AZ3CIDR",
            "PrivateSubnet3AZ1CIDR",
            "PrivateSubnet3AZ2CIDR",
            "PrivateSubnet3AZ3CIDR",
            "PrivateSubnet4AZ1CIDR",
            "PrivateSubnet4AZ2CIDR",
            "PrivateSubnet4AZ3CIDR",
            "PrivateSubnet5AZ1CIDR",
            "PrivateSubnet5AZ2CIDR",
            "PrivateSubnet5AZ3CIDR",
            "PrivateSubnet6AZ1CIDR",
            "PrivateSubnet6AZ2CIDR",
            "PrivateSubnet6AZ3CIDR",
            "PrivateSubnet7AZ1CIDR",
            "PrivateSubnet7AZ2CIDR",
            "PrivateSubnet7AZ3CIDR",
            "PrivateSubnet8AZ1CIDR",
            "PrivateSubnet8AZ2CIDR",
            "PrivateSubnet8AZ3CIDR",
            "PrivateSubnet9AZ1CIDR",
            "PrivateSubnet9AZ2CIDR",
            "PrivateSubnet9AZ3CIDR",
```

```
          "PrivateSubnet10AZ1CIDR",
          "PrivateSubnet10AZ2CIDR",
          "PrivateSubnet10AZ3CIDR"
        ]
      },
      "additionalProperties": false,
      "required": [
        "VPCCIDR",
        "NumberOfAZs",
        "PrivateSubnet1AZ1CIDR",
        "PrivateSubnet1AZ2CIDR"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "VpcName",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "VpcName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1k3oui719dcju

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Lambda execution role

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Lambda Execution Role",
  "description": "Create an Lambda execution role to use with Lambda Function. Each
 ARN specified in the parameters creates a part of the IAM policy. Use the Preview
 option to see what the completed, generated, policy looks like before it is created
 and implemented.",
```

```
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleCreateIAMRole-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleCreateIAMRole-Admin"
      ],
      "default": "AWSManagedServices-HandleCreateIAMRole-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
```

```
      "properties": {
        "ServicePrincipal": {
          "description": "Must be lambda.amazonaws.com. This establishes the trust
 relationship with the Lambda service for this role.",
          "type": "string",
          "enum": [
            "lambda.amazonaws.com"
          ],
          "default": "lambda.amazonaws.com"
        },
        "RoleName": {
          "description": "A name for the IAM role. The name can be up to 64 characters
 in length and is limited to use characters a-z, A-Z, 0-9, and _+=,.@-.",
          "type": "string",
          "pattern": "^(?![aA][mMwW][sS]|customer-mc|managementhost|ms-)[a-zA-Z0-9_
+=,.@-]{1,64}$"
        },
        "RolePath": {
          "description": "A path for the IAM role, a string of characters consisting of
 either a forward slash (/) by itself or a string that must begin and end with forward
 slash (/).",
          "type": "string",
          "default": "/",
          "pattern": "^\\/{1}([^\\/]*\\/)?$"
        },
        "Preview": {
          "description": "Yes to preview the IAM role policy created with the specified
 parameter values, without creating the role; No to not preview it but to create and
 implement the role. The preview is provided as a JSON in the execution output. In
 order to implement the policy after preview, create a copy of the RFC and set the
 Preview parameter to No, then submit.",
          "type": "string",
          "default": "No",
          "enum": [
            "Yes",
            "No"
          ]
        },
        "LambdaFunctionArns": {
          "description": "A list of Amazon resource names (ARNs) of Lambda functions.
 Scopes down the policy for read/write access to default CloudWatch log groups for
 Lambda functions.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):lambda:[a-z0-9-]+:[0-9]{12}:function:.
+)$|^$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "VPCAccess": {
          "description": "Yes to connect your function to the account VPC to access
 private resources while the function is running. No to not connect your function
 to the account VPC. For details, see the AWS documentation on configuring a Lambda
 function.",
          "type": "string",
          "default": "No",
          "enum": [
            "Yes",
            "No"
          ]
        },
        "S3ReadAccess": {
          "description": "A list of Amazon resource names (ARNs) of S3 buckets. Scopes
 down the policy for S3 read access to the given buckets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(^arn:(aws|aws-us-gov):s3:::.+$)|(^$)"
          },
          "maxItems": 50
        },
        "S3WriteAccess": {
          "description": "A list of S3 bucket ARNs. Scopes down the policy for S3 write
 access to the given buckets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(^arn:(aws|aws-us-gov):s3:::.+$)|^[*]$|(^$)"
          },
          "maxItems": 50
        },
        "KMSReadAccess": {
          "description": "A list of KMS key ARNs. Scopes down the policy for KMS read
 access to the given KMS keys only.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:(key|alias)/.
+)$|^$"
          },
          "maxItems": 50
        },
        "KMSCryptographicOperationAccess": {
          "description": "A list of KMS key ARNs. Scopes down the policy for
 cryptographic operation access to the given ARNs only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/[a-f0-9]{8}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
          },
          "maxItems": 50
        },
        "SSMReadAccess": {
          "description": "A list of SSM parameter ARNs. Scopes down the policy for SSM
 read access to the given parameters only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):ssm:[a-z0-9-]+:[0-9]{12}:parameter/.+)$|
^$"
          },
          "maxItems": 50
        },
        "SSMWriteAccess": {
          "description": "A list of SSM parameter ARNs. Scopes down the policy for SSM
 write access to given parameters only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):ssm:[a-z0-9-]+:[0-9]{12}:parameter/.+)$|
^$"
          },
          "maxItems": 50
        },
        "CloudWatchLogsReadAccess": {
          "description": "A list of CloudWatch resource ARNs. Scopes down the policy
 for read access to given CloudWatch Logs resource only.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):logs:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^
$"
          },
          "maxItems": 50
        },
        "CloudWatchLogsWriteAccess": {
          "description": "A list of CloudWatch resource ARNs. Scopes down the policy
 for write access to given CloudWatch Logs resource only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):logs:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchAlarmReadAccess": {
          "description": "A list of CloudWatch alarm ARNs. Scopes down the policy for
 read access to given CloudWatch alarms only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):cloudwatch:[a-z0-9-]+:[0-9]{12}:alarm:.
+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchAlarmWriteAccess": {
          "description": "A list of CloudWatch alarm ARNs. Scopes down the policy for
 write access to given CloudWatch alarms only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):cloudwatch:[a-z0-9-]+:[0-9]{12}:alarm:.
+)$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchMetricsReadAccess": {
          "description": "For read access to metrics, use an asterisk ( * ). Scopes
 down the policy for read access to all CloudWatch metrics.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^[*]$|^$"
          },
          "maxItems": 50
        },
        "CloudWatchMetricsWriteAccess": {
          "description": "A list of CloudWatch metric namespaces. Scopes down the
 policy for write access to given CoudWatch metric namespaces only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(.*?)|^$"
          },
          "maxItems": 50
        },
        "SecretsManagerReadAccess": {
          "description": "A list of Secrets Manager secret ARNs. Scopes down the policy
 for read access to given secrets only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):secretsmanager:[a-z0-9-]+:[0-9]
{12}:secret:.+)$|^$"
          },
          "maxItems": 50
        },
        "SNSReadAccess": {
          "description": "A list of SNS resource ARNs. Scopes down the policy for SNS
 read access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sns:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^$"
          },
          "maxItems": 50
        },
        "SNSWriteAccess": {
          "description": "A list of SNS resource ARNs. Scopes down the policy for SNS
 write access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sns:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
```

```
        },
        "SQSReadAccess": {
          "description": "A list of SQS resource ARNs. Scopes down the policy for SQS
 read access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sqs:[a-z0-9-]+:[0-9]{12}:.+)$|^[*]$|^$"
          },
          "maxItems": 50
        },
        "SQSWriteAccess": {
          "description": "A list of SQS resource ARNs. Scopes down the policy for SQS
 write access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):sqs:[a-z0-9-]+:[0-9]{12}:.+)$|^$"
          },
          "maxItems": 50
        },
        "DynamoDBResourceReadAccess": {
          "description": "A list of DynamoDB resource ARNs. Scopes down the policy for
 DynamoDB read access to given resources only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):dynamodb:[a-z0-9-]+:[0-9]{12}:.+)$|
^[*]$|^$"
          },
          "maxItems": 50
        },
        "DynamoDBDataReadWriteAccess": {
          "description": "A list of DynamoDB table ARNs. Scopes down the policy for
 DynamoDB data read and write access to given tables only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):dynamodb:[a-z0-9-]+:[0-9]{12}:table/.
+)$|^$"
          },
          "maxItems": 50
        },
        "LambdaReadAccess": {
```

```
          "description": "A list of Lambda function arns. Scopes down the policy for
 read access to given Lambda functions only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):lambda:[a-z0-9-]+:[0-9]{12}:function:.
+)$|^$"
          },
          "maxItems": 50
        },
        "LambdaInvokeAccess": {
          "description": "A list of Lambda function arns. Scopes down the policy for
 invoke access to given Lambda functions only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):lambda:[a-z0-9-]+:[0-9]{12}:function:.
+)$|^$"
          },
          "maxItems": 50
        },
        "EventsReadAccess": {
          "description": "A list of EventBridge event bus, rule arns or both. Scopes
 down the policy for read access to given EventBridge event bus, rule arns or both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):events:[a-z0-9-]+:[0-9]{12}:(event-bus|
rule)/.+)$|^$"
          },
          "maxItems": 50
        },
        "EventsWriteAccess": {
          "description": "A list of EventBridge event bus, rule arns or both. Scopes
 down the policy for write access to given EventBridge event bus, rule arns or both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):events:[a-z0-9-]+:[0-9]{12}:(event-bus|
rule)/.+)$|^$"
          },
          "maxItems": 50
        },
        "STSAssumeRole": {
```

```
          "description": "A list of IAM role ARNs. Scopes down the policy for STS
assume role to given IAM roles only.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-us-gov):iam::[0-9]{12}:role/.+)$|^$"
          },
          "maxItems": 50
        },
        "AdditionalPolicy": {
          "description": "An additional policy document, as a JSON that is less
permissive than the AMS baseline policy. For details on AMS baseline policy see AMS
documentation.",
          "type": "string",
          "pattern": "^[\\s\\S]*$",
          "maxLength": 10240
        }
      },
      "metadata": {
        "ui:order": [
          "ServicePrincipal",
          "RoleName",
          "RolePath",
          "Preview",
          "LambdaFunctionArns",
          "VPCAccess",
          "S3ReadAccess",
          "S3WriteAccess",
          "KMSReadAccess",
          "KMSCryptographicOperationAccess",
          "SSMReadAccess",
          "SSMWriteAccess",
          "CloudWatchLogsReadAccess",
          "CloudWatchLogsWriteAccess",
          "CloudWatchAlarmReadAccess",
          "CloudWatchAlarmWriteAccess",
          "CloudWatchMetricsReadAccess",
          "CloudWatchMetricsWriteAccess",
          "SecretsManagerReadAccess",
          "SNSReadAccess",
          "SNSWriteAccess",
          "SQSReadAccess",
          "SQSWriteAccess",
          "DynamoDBResourceReadAccess",
```

```
          "DynamoDBDataReadWriteAccess",
          "LambdaReadAccess",
          "LambdaInvokeAccess",
          "EventsReadAccess",
          "EventsWriteAccess",
          "STSAssumeRole",
          "AdditionalPolicy"
        ]
      },
      "required": [
        "ServicePrincipal",
        "RoleName",
        "LambdaFunctionArns",
        "Preview",
        "VPCAccess"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1ksyoxreh35tu

Classifications:

- Deployment | Managed landing zone | Management account | Create custom OUs

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
  "name": "Create Custom OUs",
  "description": "Create multiple custom AWS organizational units (OU) under the
 following paths, \"customer-managed\", \"applications:managed\", \"applications:tools
\" and \"applications:development\".",
  "type": "object",
  "properties": {
    "CustomOUPaths": {
      "description": "The OU path to create. For example: customer-
managed:ActiveDirectory or applications:managed:SAP. There is a maximum of five
 nested OUs starting from the first OU, and you can only create 10 OUs per RFC. For
 information on creating an OU path, please refer to AWS documentation.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 10,
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "CustomOUPaths"
    ]
  },
  "additionalProperties": false,
  "required": [
    "CustomOUPaths"
  ]
}
```


Schema for Change Type ct-1malj7snzxrkr

Classifications:

- Deployment | Advanced stack components | Redshift | Create (cluster)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create an Amazon Redshift cluster",
  "description": "Create an Amazon Redshift cluster that is a fully managed data
 warehouse that consists of a set of compute nodes.",
```

```
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "minLength": 1,
           "maxLength": 255
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Key",
           "Value"
         ]
```

```
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-n8kpln6rtg1eiq83b",
      "type": "string",
      "enum": [
        "stm-n8kpln6rtg1eiq83b"
      ],
      "default": "stm-n8kpln6rtg1eiq83b"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterIdentifier": {
          "type": "string",
          "description": "A unique identifier for the cluster.",
          "pattern": "^$|^[a-z]+(-?[a-z0-9]+)+$",
          "default": "",
          "minLength": 0,
          "maxLength": 63
        },
        "ClusterType": {
          "type": "string",
          "description": "The type of cluster. On a single-node cluster, the node is
shared for leader and compute functionality. On a multi-node cluster, the leader node
is separate from the compute nodes.",
          "enum": [
```

```
            "single-node",
            "multi-node"
          ],
          "default": "multi-node"
        },
        "IamRoles": {
          "type": "string",
          "description": "A comma delimited list of up to 10 AWS Identity and
 Access Management (IAM) roles that the cluster can use to access other AWS
 services. Supply the IAM roles by their Amazon Resource Name (ARN), in the form
 arn:aws:iam::000000000000:role/customer_redshift_role. The role name must be prefixed
 with \"customer\". Leave blank to not attach any roles to the cluster.",
          "pattern": "^(arn:aws:iam::[0-9]{12}:role/customer[\\w-]+)(,arn:aws:iam::
[0-9]{12}:role/customer[\\w-]+){0,9}$|^$",
          "default": ""
        },
        "ParameterGroupName": {
          "type": "string",
          "description": "The name of an existing Amazon Redshift parameter group.",
          "default": ""
        },
        "NumberOfNodes": {
          "type": "string",
          "description": "The number of compute nodes in the cluster. Only applicable
 if ClusterType = multi-mode.",
          "pattern": "^([2-9]|[1-8][0-9]|9[0-9]|100)$|^$",
          "default": "2"
        },
        "NodeType": {
          "type": "string",
          "description": "The type of an Amazon Redshift cluster node. The node type
 determines the CPU, RAM, storage capacity, and storage drive type for each node.",
          "enum": [
            "ds2.xlarge",
            "ds2.8xlarge",
            "dc2.large",
            "dc2.8xlarge",
            "dc1.large",
            "dc1.8xlarge",
            "ra3.4xlarge",
            "ra3.16xlarge"
          ],
          "default": "dc2.large"
        },
```

```
        "ClusterSubnetGroup": {
          "type": "string",
          "description": "The name of an existing Amazon Redshift subnet group.",
          "pattern": "^[a-zA-Z0-9._-]{1,255}$"
        },
        "DatabaseName": {
          "type": "string",
          "description": "The name of the first database to be created when the cluster
is created.",
          "pattern": "^[a-zA-Z0-9]{1,64}$"
        },
        "MasterUsername": {
          "type": "string",
          "description": "The name that you use with the configured MasterUserPassword
to log in to an Amazon Redshift cluster. Must begin with a letter and contain from 1
to 128 alphanumeric characters.",
          "pattern": "^[a-zA-Z][a-zA-Z0-9]{0,127}$"
        },
        "MasterUserPassword": {
          "type": "string",
          "description": "The password that you use with the configured MasterUsername
to log in to an Amazon Redshift cluster. Must contain from 8 to 64 printable ASCII
characters including at least one uppercase letter, one lowercase letter, and one
decimal digit. It cannot contain backslash, forwardslash, single or double quotes, at
sign, or whitespace.",
          "pattern": "^(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])[^ \"@'/\\\\]{8,64}$",
          "maxLength": 64,
          "minLength": 8,
          "metadata": {
            "ams:sensitive": true
          }
        },
        "AllowVersionUpgrade": {
          "type": "string",
          "description": "True to apply upgrades to the engine that is running on the
cluster, during the maintenance window; false to not.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "SecurityGroups": {
          "type": "array",
```

```
          "description": "The identifiers of the security groups to control traffic to
 and from the Redshift cluster.",
          "items": {
            "type": "string",
            "pattern": "^sg-(?=.*[a-z])(?=.*[0-9])(?:.{8}|.{17})$|^$",
            "default": ""
          },
          "uniqueItems": true
        },
        "DatabasePortNumber": {
          "type": "integer",
          "description": "The port number on which the cluster accepts incoming
 connections.",
          "default": 5439,
          "minimum": 1150,
          "maximum": 65535
        },
        "AutomatedSnapshotRetentionPeriod": {
          "type": "integer",
          "description": "The number of days that automated snapshots are retained. The
 default is to retain 7 days of snapshots, and the maximum value is 35 days. To disable
 automated snapshot, use 0.",
          "default": 7,
          "minimum": 0,
          "maximum": 35
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
          "description": "The weekly time range (in UTC) during which automated cluster
 maintenance can occur. The format of the time range is ddd:hh24:mi-ddd:hh24:mi. Leave
 blank to allow Amazon Redshift to choose the suitable maintenance window.",
          "pattern": "^[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$|^$",
          "default": ""
        },
        "KmsKeyId": {
          "type": "string",
          "description": "The ID of the AWS Key Management Service (AWS KMS) key that
 you want to use to encrypt data in the cluster. Leave blank to not encrypt data.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
          "default": ""
        }
      },
      "metadata": {
```

```
        "ui:order": [
          "ClusterIdentifier",
          "DatabaseName",
          "DatabasePortNumber",
          "MasterUsername",
          "MasterUserPassword",
          "NodeType",
          "ClusterType",
          "NumberOfNodes",
          "ParameterGroupName",
          "ClusterSubnetGroup",
          "SecurityGroups",
          "AllowVersionUpgrade",
          "AutomatedSnapshotRetentionPeriod",
          "PreferredMaintenanceWindow",
          "IamRoles",
          "KmsKeyId"
        ]
      },
      "required": [
        "ClusterSubnetGroup",
        "DatabaseName",
        "MasterUsername",
        "MasterUserPassword"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
```

```
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1n323w7eu27u9

Classifications:

- Management | Advanced stack components | Redshift | Pause cluster

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Pause Redshift Cluster",
  "description": "Pause an Amazon Redshift cluster. If a recent snapshot is not
 available, a temporary manual snapshot is created with a retention period of one day.
 This snapshot is deleted towards the end of execution for both success and failure
 scenarios. It is safe for AMS to delete this snapshot as pausing the cluster creates
 an automated snapshot by default.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-PauseRedshiftCluster.",
      "type": "string",
      "enum": [
        "AWSManagedServices-PauseRedshiftCluster"
      ],
      "default": "AWSManagedServices-PauseRedshiftCluster"
    },
    "Region": {
      "description": "The AWS Region in which the Amazon Redshift cluster is located,
 in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterIdentifier": {
          "description": "The Amazon Redshift cluster identifier. For example, myred-
cluster-1.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$",
            "minLength": 1,
            "maxLength": 63
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ClusterIdentifier"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ClusterIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1n9gfnog5x7fl

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create entity or policy (read-write permissions)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Entity or Policy (read-write permissions)",
  "description": "Create Identity and Access Management (IAM) role or policy with read-write permissions. You must have enabled this feature with change type ct-1706xvvk6j9hf before submitting this request. Automated IAM provisioning with read-write permissions runs over 200 validations to help ensure successful outcomes.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin"
      ],
      "default": "AWSManagedServices-HandleAutomatedIAMProvisioningCreate-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
```

```
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ValidateOnly": {
          "description": "Yes to only validate the IAM entity or policy with the
specified parameter values, without creating the entity or policy; No to validate
and create the entity or policy. The validation result is provided as a JSON in the
execution output. In order to implement after validation, create a copy of the RFC and
set the ValidateOnly parameter to No, then submit.",
          "type": "string",
          "enum": [
            "Yes",
            "No"
          ],
          "default": "No"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "ValidateOnly"
        ]
      },
      "required": [
        "ValidateOnly"
      ]
    },
    "RoleDetails": {
      "type": "object",
      "properties": {
        "Roles": {
          "description": "Add a role.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "RoleName": {
```

```
                "description": "A name for the IAM role. The name can be up to 64
 characters in length, and is limited to use characters a-z, A-Z, 0-9, hyphen and
 underscore.",
                "type": "string",
                "pattern": "^[a-zA-Z0-9_-]{1,64}$"
              },
              "Description": {
                "description": "A meaningful description for the role.",
                "type": "string",
                "minLength": 0,
                "maxLength": 5200,
                "default": ""
              },
              "AssumeRolePolicyDocument": {
                "description": "A JSON policy document that you want to associate with
 the role, defining which entities can assume the role. This is known as the Assume
 role policy. Paste the contents into the input.",
                "type": "string",
                "minLength": 2,
                "maxLength": 131072
              },
              "ManagedPolicyArns": {
                "description": "A list of Amazon Resource Names (ARNs) of the IAM
 managed policies that you want to attach to the role. Both AWS managed policies
 and customer managed policies are allowed. If you create a managed policy in
 this RFC and wish to attach to this role then list the policy here in the form
 arn:aws:iam::AccountId:policy/NameOfYourPolicy.",
                "type": "array",
                "items": {
                  "type": "string",
                  "pattern": "^arn:[\\w+=/,.@-]+:iam::[0-9]{12}:policy(/[\\w+=/,.@-]+)?
$|^arn:[\\w+=/,.@-]+:iam::aws:policy(/[\\w+=/,.@-]+)?$"
                },
                "minItems": 0,
                "maxItems": 20
              },
              "Path": {
                "description": "A path for the IAM role, a string of characters
 consisting of either a forward slash (/) by itself or a string that must begin and end
 with forward slash (/).",
                "type": "string",
                "default": "/",
                "pattern": "^\\/{1}([^\\/]*\\/)?$|^$",
                "minLength": 0,
```

```
                "maxLength": 512
              },
              "MaxSessionDuration": {
                "description": "The maximum session duration (in seconds) that you want
 to set for the specified role. If you do not specify a value for this setting, the
 default value of one hour is applied. This setting can have a value from 1 hour to 4
 hours. The MaxSessionDuration time begins with the assumption of the role.",
                "type": "string",
                "default": "3600",
                "pattern": "^(360\\d|36[1-9]\\d|3[7-9]\\d{2}|[4-9]\\d{3}|1[0-3]\\d{3}|
14[0-3]\\d{2}|14400)$"
              },
              "PermissionsBoundary": {
                "description": "The ARN of the policy used to set the permissions
 boundary for the role. A permissions boundary uses a managed policy to set the maximum
 permissions that an identity-based policy can grant to an IAM entity.",
                "type": "string",
                "default": "",
                "pattern": "^$|^arn:[\\w+=/,.@-]+:iam::[0-9]{12}:policy(/[\\w
+=/,.@-]+)?$"
              },
              "InstanceProfile": {
                "description": "Yes to create an instance profile and associate the
 role with it. No to not create an instance profile.",
                "type": "string",
                "default": "No",
                "enum": [
                  "Yes",
                  "No"
                ]
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "RoleName",
                "Description",
                "AssumeRolePolicyDocument",
                "ManagedPolicyArns",
                "Path",
                "MaxSessionDuration",
                "PermissionsBoundary",
                "InstanceProfile"
              ]
```

```
            },
            "required": [
              "RoleName",
              "AssumeRolePolicyDocument"
            ]
          },
          "minItems": 0,
          "maxItems": 1,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Roles"
        ]
      }
    },
    "ManagedPolicyDetails": {
      "type": "object",
      "properties": {
        "Policies": {
          "description": "Add a customer managed policy. To attach a policy to a role
 created in this RFC, provide the policy in ARN format (arn:aws:iam::AccountId:policy/
NameOfYourPolicy) in the ManagedPolicyArns field of the role. Alternatively, use
 ct-1e0xmuy1diafq to update the role and attach the policy.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "ManagedPolicyName": {
                "description": "A name for the IAM policy. The name can be up to 122
 characters in length, and is limited to use characters a-z, A-Z, 0-9, hyphen and
 underscore.",
                "type": "string",
                "pattern": "^[a-zA-Z0-9_-]{1,122}$"
              },
              "Description": {
                "description": "A meaningful description for the policy.",
                "type": "string",
                "minLength": 0,
                "maxLength": 5200,
                "default": ""
              },
```

```
            "Path": {
              "description": "A path for the policy, a string of characters
consisting of either a forward slash (/) by itself or a string that must begin and end
with forward slash (/).",
              "type": "string",
              "default": "/",
              "pattern": "^\\/{1}([^\\/]*\\/)?$|^$",
              "minLength": 0,
              "maxLength": 512
            },
            "PolicyDocument": {
              "description": "The JSON policy document that you want to use as the
content for the new policy. Paste the content into the input field.",
              "type": "string",
              "minLength": 2,
              "maxLength": 131072
            }
          },
          "additionalProperties": false,
          "metadata": {
            "ui:order": [
              "ManagedPolicyName",
              "Description",
              "Path",
              "PolicyDocument"
            ]
          },
          "required": [
            "ManagedPolicyName",
            "PolicyDocument"
          ]
        },
        "minItems": 0,
        "maxItems": 1,
        "uniqueItems": true
      }
    },
    "additionalProperties": false,
    "metadata": {
      "ui:order": [
        "Policies"
      ]
    }
  }
```

```
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters",
      "RoleDetails",
      "ManagedPolicyDetails"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1o1x2itfd6rk8

Classifications:

- Management | Advanced stack components | EC2 instance stack | Update (with additional volumes)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update EC2 stack (with additional volumes)",
  "description": "Use to modify the properties of an EC2 instance created using CT id
 ct-1aqsjf86w6vxg, version 3.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC that contains the EC2 Instance, in the form
 vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the EC2 instance with additional volumes that you
 are updating, in the form stack-a1b2c3d4e5f67890e.",
```

```
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the EC2 instance with additional
 volumes.",
      "type": "object",
      "properties": {
        "InstanceDetailedMonitoring": {
          "description": "True to enable detailed monitoring on the instance, false to
 use only basic monitoring.",
          "type": "boolean"
        },
        "InstanceEBSOptimized": {
          "description": "True for the instance to be optimized for Amazon Elastic
 Block Store I/O, false for it to not be. If you set this to true, choose an
 InstanceType that supports EBS optimization. Updates will stop and start Amazon EBS-
backed instances.",
          "type": "boolean"
        },
        "InstanceProfile": {
          "description": "An IAM instance profile name defined in your account for the
 EC2 instance.",
          "type": "string",
          "minLength": 1,
          "maxLength": 128,
          "pattern": "^customer[\\w-]{1,120}$"
        },
        "InstanceSecondaryPrivateIpAddressCount": {
          "description": "The number of secondary private IP addresses that EC2
 automatically assigns to the primary network interface. The number of secondary IP
 addresses that can be assigned is dependent on the type of instance used.",
          "type": "integer",
          "minimum": 0
        },
        "InstanceTerminationProtection": {
          "description": "True to prevent the instance from being terminated through
 the API, false to allow it. Termination protection must be disabled before deleting
 the stack or performing an update where instance replacement is required, otherwise
 failures will occur.",
          "type": "boolean"
        },
        "InstanceType": {
```

```
          "description": "The type of EC2 instance to deploy. If InstanceEBSOptimized
 = true, specify an InstanceType that supports EBS optimization. Changing the instance
 type will result in instance stop and start.",
          "type": "string"
        },
        "InstanceUserData": {
          "description": "A newline-delimited string where each line is part of the
 script to be run on boot. Changing the UserData will result in instance stop and
 start. Note: Existing instances do not pick up changes in UserData automatically,
 in order for the instance to execute modified UserData you must perform additional
 changes by logging in to the instance.",
          "type": "string",
          "maxLength": 4096
        },
        "Volume1Iops": {
          "type": "integer",
          "description": "The Iops to use for Volume1 if Volume1Type = io1.",
          "minimum": 0,
          "maximum": 32000
        },
        "Volume1KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume1. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume1. Updates are not supported. Use only if Volume1 is a new volume.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume1Name": {
          "type": "string",
          "description": "The device name for Volume1 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume1. Leave blank to skip creation of Volume1. Updates are not supported.
 Use only if Volume1 is a new volume."
        },
        "Volume1Size": {
          "type": "integer",
          "description": "The size of Volume1 in GiB. Only size increases are supported
 when resizing.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume1Snapshot": {
          "type": "string",
```

```
          "description": "Snapshot ID for Volume1. Updates are not supported. Use only
 if Volume1 is a new volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume1Type": {
          "type": "string",
          "description": "The volume type for Volume1. Choose io1 or gp2 for SSD-
backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-backed
 volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "gp2",
            "sc1",
            "st1"
          ]
        },
        "Volume2Iops": {
          "type": "integer",
          "description": "The Iops to use for Volume2 if Volume2Type = io1.",
          "minimum": 0,
          "maximum": 32000
        },
        "Volume2KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume2. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume2. Updates are not supported. Use only if Volume2 is a new volume.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume2Name": {
          "type": "string",
          "description": "The device name for Volume2 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume2. Leave blank to skip creation of Volume2. Updates are not supported.
 Use only if Volume2 is a new volume."
        },
        "Volume2Size": {
          "type": "integer",
          "description": "The size of Volume2 in GiB. Only size increases are supported
 when resizing.",
          "minimum": 1,
```

```
          "maximum": 16384
        },
        "Volume2Snapshot": {
          "type": "string",
          "description": "Snapshot ID for Volume2. Updates are not supported. Use only
 if Volume2 is a new volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume2Type": {
          "type": "string",
          "description": "The volume type for Volume2. Choose io1 or gp2 for SSD-
backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-backed
 volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "gp2",
            "sc1",
            "st1"
          ]
        },
        "Volume3Iops": {
          "type": "integer",
          "description": "The Iops to use for Volume3 if Volume3Type = io1.",
          "minimum": 0,
          "maximum": 32000
        },
        "Volume3KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume3. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume3. Updates are not supported. Use only if Volume3 is a new volume.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume3Name": {
          "type": "string",
          "description": "The device name for Volume3 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume3. Leave blank to skip creation of Volume3. Updates are not supported.
 Use only if Volume3 is a new volume."
        },
        "Volume3Size": {
```

```
          "type": "integer",
          "description": "The size of Volume3 in GiB. Only size increases are supported
 when resizing.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume3Snapshot": {
          "type": "string",
          "description": "Snapshot ID for Volume3. Updates are not supported. Use only
 if Volume3 is a new volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume3Type": {
          "type": "string",
          "description": "The volume type for Volume3. Choose io1 or gp2 for SSD-
backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-backed
 volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "gp2",
            "sc1",
            "st1"
          ]
        },
        "Volume4Iops": {
          "type": "integer",
          "description": "The Iops to use for Volume4 if Volume4Type = io1.",
          "minimum": 0,
          "maximum": 32000
        },
        "Volume4KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume4. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume4. Updates are not supported. Use only if Volume4 is a new volume.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "Volume4Name": {
          "type": "string",
          "description": "The device name for Volume4 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
```

```
 to create Volume4. Leave blank to skip creation of Volume4. Updates are not supported.
 Use only if Volume4 is a new volume."
        },
        "Volume4Size": {
          "type": "integer",
          "description": "The size of Volume4 in GiB. Only size increases are supported
 when resizing.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume4Snapshot": {
          "type": "string",
          "description": "Snapshot ID for Volume4. Updates are not supported. Use only
 if Volume4 is a new volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume4Type": {
          "type": "string",
          "description": "The volume type for Volume4. Choose io1 or gp2 for SSD-
backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-backed
 volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "gp2",
            "sc1",
            "st1"
          ]
        },
        "Volume5Iops": {
          "type": "integer",
          "description": "The Iops to use for Volume5 if Volume5Type = io1.",
          "minimum": 0,
          "maximum": 32000
        },
        "Volume5KmsKeyId": {
          "type": "string",
          "description": "ID or ARN of the KMS master key to be used to encrypt
 Volume5. Specify default to use the default EBS KMS Key. Leave blank to not encrypt
 Volume5. Updates are not supported. Use only if Volume5 is a new volume.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
```

```
        "Volume5Name": {
          "type": "string",
          "description": "The device name for Volume5 (for example, /dev/sdf through /
dev/sdp for Linux or xvdf through xvdp for Windows). A valid value for this is required
 to create Volume5. Leave blank to skip creation of Volume5. Updates are not supported.
 Use only if Volume5 is a new volume."
        },
        "Volume5Size": {
          "type": "integer",
          "description": "The size of Volume5 in GiB. Only size increases are supported
 when resizing.",
          "minimum": 1,
          "maximum": 16384
        },
        "Volume5Snapshot": {
          "type": "string",
          "description": "Snapshot ID for Volume5. Updates are not supported. Use only
 if Volume5 is a new volume.",
          "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$|^$"
        },
        "Volume5Type": {
          "type": "string",
          "description": "The volume type for Volume5. Choose io1 or gp2 for SSD-
backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-backed
 volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "standard",
            "io1",
            "gp2",
            "sc1",
            "st1"
          ]
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "InstanceDetailedMonitoring",
          "InstanceEBSOptimized",
          "InstanceProfile",
          "InstanceType",
          "InstanceUserData",
          "InstanceSecondaryPrivateIpAddressCount",
```

```
          "InstanceTerminationProtection",
          "Volume1Name",
          "Volume1Size",
          "Volume1Type",
          "Volume1KmsKeyId",
          "Volume1Iops",
          "Volume1Snapshot",
          "Volume2Name",
          "Volume2Size",
          "Volume2Type",
          "Volume2KmsKeyId",
          "Volume2Iops",
          "Volume2Snapshot",
          "Volume3Name",
          "Volume3Size",
          "Volume3Type",
          "Volume3KmsKeyId",
          "Volume3Iops",
          "Volume3Snapshot",
          "Volume4Name",
          "Volume4Size",
          "Volume4Type",
          "Volume4KmsKeyId",
          "Volume4Iops",
          "Volume4Snapshot",
          "Volume5Name",
          "Volume5Size",
          "Volume5Type",
          "Volume5KmsKeyId",
          "Volume5Iops",
          "Volume5Snapshot"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
```

```
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1opjmhuddw194

Classifications:

- Management | Managed landing zone | Management account | Enable developer mode

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Developer Mode",
  "description": "Enable Developer Mode for an existing application account. Note that,
 in Developer mode, you are responsible for monitoring infrastructure resources that
 are provisioned outside of the AMS change management process.",
  "type": "object",
  "properties": {
    "ApplicationAccountId": {
      "description": "The account ID of the application account to have Developer mode
 enabled.",
      "type": "string",
      "pattern": "^[0-9]{12}$"
    }
  },
  "metadata": {
    "ui:order": [
      "ApplicationAccountId"
    ]
  },
  "additionalProperties": false,
  "required": [
    "ApplicationAccountId"
  ]
}
```

Schema for Change Type ct-1oxx2g2d7hc90

Classifications:

- Deployment | Advanced stack components | Security group | Create (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Security Group (review required)",
  "description": "Create a security group, and optionally associate it with AWS
 resources.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "The ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the security group. The name can be up to 255
 characters in length, and is limited to these characters a-z, A-Z, 0-9, spaces,
 and ._-:/()#,@[]+=&;{}!$*. The name cannot start with \"sg-\", and must be unique
 within the VPC.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Description": {
      "description": "Meaningful information about the security group. The description
 can be up to 255 characters in length, and is limited to these characters a-z, A-Z,
 0-9, spaces, and ._-:/()#,@[]+=&;{}!$*.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "AssociatedResources": {
      "description": "AWS resources to associate the security group to. For example,
 EC2 instance IDs, RDS DB instance IDs, Load Balancer names, DSM replication instance
 names, EFS mount target IDs, ElastiCache cluster IDs.",
      "type": "array",
      "items": {
```

```
        "type": "string",
        "minLength": 1,
        "maxLength": 64
      },
      "minItems": 0,
      "maxItems": 10,
      "uniqueItems": true
    },
    "InboundRules": {
      "description": "Inbound rules for the security group. No inbound traffic
 originating from another host to your instance is allowed until you add inbound rules
 to the security group.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
            "description": "A port number or a port range. For example, 80 or
 49152-65535. For a port range of all ports, specify -1.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
          },
          "Source": {
            "description": "An IP address, or an IP address range in CIDR notation (for
 example, 203.0.113.5/32), or the ID of another security group in the same region.
 To use this security group, specify self. From behind a firewall, use the public IP
 address or range used by the client computers.",
            "type": "string",
            "minLength": 1,
            "maxLength": 64
          },
          "Description": {
            "description": "A meaningful description of the inbound rule.",
            "type": "string",
```

```
            "minLength": 0,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Source",
            "Description"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Source"
        ]
      },
      "minItems": 0,
      "maxItems": 50
    },
    "OutboundRules": {
      "description": "Outbound rules for the security group. No outbound traffic
 originating from your instance is allowed until you add outbound rules.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
            "description": "A port number or a port range. For example, 80 or
 49152-65535. For a port range of all ports, specify -1.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
```

```
          },
          "Destination": {
            "description": "An IP address, or an IP address range in CIDR notation (for
example, 203.0.113.5/32), or the ID of another security group in the same region.
To use this security group, specify self. From behind a firewall, use the public IP
address or range used by the client computers.",
            "type": "string",
            "minLength": 1,
            "maxLength": 64
          },
          "Description": {
            "description": "A meaningful description of the outbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Destination",
            "Description"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Destination"
        ]
      },
      "minItems": 0,
      "maxItems": 50
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
```

```
    },
    "Tags": {
      "description": "Up to 50 tags (key/value pairs) to categorize the security
 group.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "Name",
      "Description",
      "AssociatedResources",
      "InboundRules",
```

```
      "OutboundRules",
      "Priority",
      "Tags"
    ]
  },
  "required": [
    "VpcId",
    "Name",
    "Description"
  ]
}
```

Schema for Change Type ct-1pvlhug439gl2

Classifications:

- Management | Advanced stack components | EC2 instance stack | Associate private IP addresses (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Associate Private IP Addresses",
  "description": "Associate one or more secondary private IP addresses to the specified
 network interface.",
  "type": "object",
  "properties": {
    "NetworkInterfaceId": {
      "description": "The ID of the network interface, in the form
 eni-0123456789abcdef0.",
      "type": "string",
      "pattern": "^eni-[a-f0-9]{17}"
    },
    "PrivateIpAddresses": {
      "description": "The IP addresses to be associated as a secondary private IP
 addresses to the network interface, for example, '10.0.0.82', '10.0.0.83'.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^(10(\\.(25[0-5]|2[0-4][0-9]|1[0-9]{1,2}|[0-9]{1,2})){3}|((172\
\.(1[6-9]|2[0-9]|3[01]))|192\\.168)(\\.(25[0-5]|2[0-4][0-9]|1[0-9]{1,2}|[0-9]{1,2}))
{2})$"
```

```
      },
      "minItems": 1,
      "maxItems": 50
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "NetworkInterfaceId",
      "PrivateIpAddresses",
      "Priority"
    ]
  },
  "required": [
    "NetworkInterfaceId",
    "PrivateIpAddresses"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1pybwg08h8qsz

Classifications:

- Management | Host security | Malware full system scan | Disable (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disable malware scans",
  "description": "Use to disable periodic malware full system scan feature in all EC2
 instances deployed in a single VPC.",
  "type": "object",
  "properties": {
```

```
    "VpcId": {
      "description": "ID of the VPC to disable periodic malware scans on, in the form
 of vpc-12345678 or vpc-1234567890abcdef0.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "VpcId"
  ]
}
```

Schema for Change Type ct-1q8q56cmwqj9m

Classifications:

- Management | Advanced stack components | ACM | Delete certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete an ACM Certificate",
  "description": "Delete an AWS Certificate Manager (ACM) certificate that is currently
 not in use and not managed by AMS.",
  "type": "object",
  "properties": {
```

```
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteACMCertificate.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteACMCertificate"
      ],
      "default": "AWSManagedServices-DeleteACMCertificate"
    },
    "Region": {
      "description": "The AWS Region of the ACM certificate, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CertificateARN": {
          "description": "The Amazon Resource Name (ARN) of the certificate to
 delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "required": [
        "CertificateARN"
      ],
      "metadata": {
        "ui:order": [
          "CertificateARN"
        ]
      }
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
```

```
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1r19m51jeijlk

Classifications:

- Deployment | Advanced stack components | Target Group | Create (for ALB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create target group for ALB",
  "description": "Use to create a target group for an Application Load Balancer.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
```

```
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-9c1t8maqho0os5k22",
      "type": "string",
      "enum": [
        "stm-9c1t8maqho0os5k22"
      ],
      "default": "stm-9c1t8maqho0os5k22"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
```

```
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ApplicationLoadBalancerArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the application load
 balancer in the form arn:aws:elasticloadbalancing:region:account-id:loadbalancer/app/
load-balancer-name/load-balancer-id. This is used to create CloudWatch alarms that
 trigger if the Target Group contains no healthy instances.",
          "pattern": "arn:aws:elasticloadbalancing:[a-z1-9\\-]{9,15}:[0-9]
{12}:loadbalancer/app/[a-zA-Z0-9\\-]{1,32}/[a-z0-9]+"
        },
        "HealthCheckHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$",
          "default": ""
        },
        "HealthCheckUnhealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check failure required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$",
          "default": ""
        },
        "HealthCheckInterval": {
          "type": "integer",
          "description": "The approximate interval, in seconds, between health checks.
 The supported values are 5 seconds to 300 seconds.",
          "default": 30,
          "minimum": 5,
          "maximum": 300
        },
        "HealthCheckTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, to wait for a response to
 a health check. Must be less than the value for HealthCheckInterval. The supported
 values are 2 seconds to 60 seconds.",
```

```
          "pattern": "60|[1-5]{1}[0-9]{1}|[2-9]{1}|^$",
          "default": ""
        },
        "HealthCheckTargetPath": {
          "type": "string",
          "description": "The ping path destination on the application hosts where the
 load balancer sends health check requests.",
          "default": "/"
        },
        "HealthCheckTargetPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health checks
 on targets. The default is traffic-port, which indicates the port on which each target
 receives traffic from the load balancer.",
          "pattern": "[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2]
[0-9]|6553[0-5]|traffic-port|",
          "default": ""
        },
        "HealthCheckTargetProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS"
          ],
          "default": "HTTP"
        },
        "ValidHTTPCode": {
          "type": "string",
          "description": "The HTTP codes that a healthy target application server must
 use in response to a health check. You can specify multiple values such as 200,202, or
 a range of values such as 200-499. Only applicable if HealthCheckTargetProtocol = HTTP
 or HTTPS.",
          "pattern": "^$|([2-4]{1}[0-9]{2}($|-|,))+",
          "default": "200"
        },
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": "80"
```

```
        },
        "Name": {
          "type": "string",
          "description": "A name for the target group. This name must be unique per
 account, per region.",
          "pattern": "[0-9a-zA-Z\\-]{0,32}",
          "default": ""
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS"
          ],
          "default": "HTTP"
        },
        "DeregistrationDelayTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing to
 wait before changing the state of a deregistering target from draining to unused.",
          "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})",
          "default": "300"
        },
        "SlowStartDuration": {
          "type": "string",
          "description": "The time period, in seconds, during which the load balancer
 sends a newly registered target a linearly-increasing share of the target group
 traffic.",
          "pattern": "[3-9]{1}[0-9]{1}|[1-8]{1}[0-9]{2}|900|0|",
          "default": ""
        },
        "StickinessCookieExpirationPeriod": {
          "type": "string",
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "pattern": "[1-9]{1}[0-9]{0,4}|[1-5]{1}[0-9]{5}|60[0-3]{1}[0-9]{3}|604[0-7]
{1}[0-9]{2}|604800|",
          "default": ""
        },
        "TargetType": {
          "type": "string",
```

```
          "description": "The registration type of the targets; determines how you
 specify the TargetGroup targets. If you choose instance, you specify the targets by
 instance ID. If you choose ip, you specify the targets by IP address. After you create
 a target group, you cannot change its target type.",
          "enum": [
            "instance",
            "ip"
          ],
          "default": "instance"
        },
        "Target1ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target1Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target1AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target1ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target2ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
```

```
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target2ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target3ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target3AvailabilityZone": {
          "type": "string",
```

```
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target3ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target4ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target4AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target4ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target5ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
```

```
          "default": ""
        },
        "Target5Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target5AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target5ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target6ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target6Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target6AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target6ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
```

```
            "",
            "all"
          ],
          "default": ""
        },
        "Target7ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target7Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target7AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target7ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        },
        "Target8ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target8Port": {
          "type": "string",
```

```
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target8AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target8ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ],
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "InstancePort",
          "InstanceProtocol",
          "ApplicationLoadBalancerArn",
          "DeregistrationDelayTimeout",
          "SlowStartDuration",
          "StickinessCookieExpirationPeriod",
          "HealthCheckTargetPath",
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckUnhealthyThreshold",
          "HealthCheckInterval",
          "HealthCheckTimeout",
          "ValidHTTPCode",
          "TargetType",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
```

```
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
          "Target4AvailabilityZone",
          "Target5ID",
          "Target5Port",
          "Target5AvailabilityZone",
          "Target6ID",
          "Target6Port",
          "Target6AvailabilityZone",
          "Target7ID",
          "Target7Port",
          "Target7AvailabilityZone",
          "Target8ID",
          "Target8Port",
          "Target8AvailabilityZone"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstancePort",
        "InstanceProtocol",
        "ApplicationLoadBalancerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
```

```
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1r1vbr8ahr156

Classifications:

- Management | AWS Backup | Recovery point | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Recovery Points",
  "description": "Delete one or more recovery points (snapshots) from the specified
 vault. Use this change type to delete recovery points that were manually created, and
 recovery points that were created through a backup plan, and that are older than 30
 days. The deletion of recovery points cannot be rolled back.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteRecoveryPoints.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteRecoveryPoints"
      ],
      "default": "AWSManagedServices-DeleteRecoveryPoints"
    },
    "Region": {
      "description": "The AWS Region in which the AWS Backup recovery point is located,
 in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BackupVaultName": {
          "description": "The name of the AWS Backup vault that contains the recovery
 point to delete.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RecoveryPointArns": {
          "description": "A list of up to 50 recovery points to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 50,
          "minItems": 1,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "BackupVaultName",
          "RecoveryPointArns"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BackupVaultName",
        "RecoveryPointArns"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
```

```
}
```

Schema for Change Type ct-1taxucdyi84iy

Classifications:

- Management | Managed Firewall | Outbound (Palo Alto) | Delete security policy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Security Policy",
  "description": "Delete a security policy for AMS managed Palo Alto firewall -
 Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be DeleteSecurityPolicy.",
      "type": "string",
      "enum": [
        "DeleteSecurityPolicy"
      ],
      "default": "DeleteSecurityPolicy"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityPolicyName": {
          "description": "The name of the security policy. Must start with custom-
sec-.",
          "type": "string",
          "pattern": "^custom-sec-[a-zA-Z0-9][a-zA-Z0-9-_]{0,51}$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "SecurityPolicyName"
        ]
      },
      "required": [
        "SecurityPolicyName"
      ]
```

```
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "RequestType",
      "Parameters"
    ]
  },
  "required": [
    "RequestType",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1urj94c3hdfu5

Classifications:

- Deployment | Managed landing zone | Networking account | Create application route table (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Application Account Route Table",
  "description": "Create a custom AWS Transit Gateway (TGW) route table for the
 application accounts in the networking account. By default, the route table does not
 connect to the on-premise network, but contains preset routes. To request connections
 to the on-premise network, submit a Management|Other|Other|Update change type.",
  "type": "object",
  "properties": {
    "TransitGatewayApplicationRouteTableName": {
      "description": "A meaningful name for the TGW route table.",
      "type": "string"
    },
    "AddPresetStaticRoutes": {
      "description": "True to create a route table with the default route (0.0.0.0/0)
 to the outbound (egress) VPC, and a route to the perimeter (DMZ) VPC and the shared
 services VPC. False to create an empty route domain with no routes. Default is
 true.",
      "type": "boolean",
```

```
      "default": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "TransitGatewayApplicationRouteTableName",
      "AddPresetStaticRoutes",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "TransitGatewayApplicationRouteTableName"
  ]
}
```

Schema for Change Type ct-1v9g9n30woc8h

Classifications:

- Management | Managed landing zone | Management account | Update StackSets stack (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update StackSets Stack",
  "description": "Update an existing AWS CloudFormation (CFN) StackSets stack to
 deploy, or to update, the instances of the stack.",
  "type": "object",
  "properties": {
    "CloudFormationTemplate": {
```

```
      "description": "The CFN template that you have configured to update the stack
set, copy the JSON and paste it into the field. Provide a value for either this, or
the CloudFormationTemplateS3Endpoint parameter.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?![\\s]*https?)[\\S\\s]*$",
      "maxLength": 20000
    },
    "CloudFormationTemplateS3Endpoint": {
      "description": "The S3 bucket endpoint for the CloudFormation template you want
to use. The bucket must be in the same account that you are using, or have a presigned
URL. Provide a value for either this, or the CloudFormationTemplate parameter.",
      "type": "string",
      "minLength": 1,
      "pattern": "^[\\s]*https?://[\\S]*[\\s]*$|^[\\s]*$",
      "maxLength": 2047
    },
    "Parameters": {
      "description": "Add up to sixty parameters (parameter name/value pairs) to
supply alternate values for parameters in your customized CloudFormation template.
By providing the parameters this way, you can reuse your CloudFormation template with
different parameter values when needed and can update any parameter value with the CFN
Update stack set (review required) change type (ct-1v9g9n30woc8h).",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Name": {
            "type": "string",
            "pattern": "[A-Za-z0-9]+$"
          },
          "Value": {
            "type": "string"
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Name",
            "Value"
          ]
        },
        "required": [
          "Name",
```

```
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "Description": {
      "description": "Description of the StackSets stack to be updated",
      "type": "string",
      "minLength": 1,
      "maxLength": 1024
    },
    "Name": {
      "description": "Name of the StackSets stack to be updated.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$",
      "maxLength": 128
    },
    "OuId": {
      "description": "The ID of the AWS organizational unit for the stack instances
being deployed. If you add a parent OU as a target, StackSets also adds any child OU
as targets. To deploy the StackSets stack instances in all OUs, use 'all'",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^(ou-[a-z0-9]{4,32}-[a-z0-9]{8,32}|r-[a-z0-9]{4,32}|all)$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "Region": {
      "description": "The AWS Region of the resources you're updating in the form of
us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the StackSets
stack.",
      "type": "array",
      "items": {
        "type": "object",
```

```
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-|aws:))[a-zA-Z0-9 .:+=@_/-]{1,128}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-|aws:))[a-zA-Z0-9 .:+=@_/-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
```

```
      "Name",
      "Description",
      "CloudFormationTemplate",
      "CloudFormationTemplateS3Endpoint",
      "Parameters",
      "Region",
      "OuId",
      "Tags",
      "Priority"
    ]
  },
  "required": [
    "Name",
    "Region",
    "OuId"
  ]
}
```

Schema for Change Type ct-1vbv99ko7bsrq

Classifications:

- Deployment | Monitoring and notification | SQS | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SQS",
  "description": "Use to create an Amazon Simple Queue Service instance for messages to
 be shared by system components.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
```

```
    },
    "StackTemplateId": {
      "description": "Must be stm-s1ejpr80000000000.",
      "type": "string",
      "enum": [
        "stm-s1ejpr80000000000"
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7,
      "uniqueItems": true
    },
```

```
   "TimeoutInMinutes": {
     "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
     "type": "number",
     "minimum": 0,
     "maximum": 60
   },
   "Parameters": {
     "description": "Specifications for the stack.",
     "type": "object",
     "properties": {
       "SQSDelaySeconds": {
         "description": "The time in seconds that the delivery of all messages in the
queue will be delayed.",
         "type": "number",
         "minimum": 0,
         "maximum": 900,
         "default": 0
       },
       "SQSMaximumMessageSize": {
         "description": "The limit of how many bytes a message can contain before SQS
rejects it.",
         "type": "number",
         "minimum": 1024,
         "maximum": 262144,
         "default": 262144
       },
       "SQSMessageRetentionPeriod": {
         "description": "The number of seconds SQS retains a message, from 60 (1
minute) to 1209600 (14 days).",
         "type": "number",
         "minimum": 60,
         "maximum": 1209600,
         "default": 345600
       },
       "SQSQueueName": {
         "description": "A name for the queue, case sensitive.",
         "type": "string",
         "pattern": "^[a-zA-Z0-9-_]{1,80}$",
         "minLength": 1,
         "maxLength": 80
       },
       "SQSReceiveMessageWaitTimeSeconds": {
```

```
          "description": "The number of seconds that the ReceiveMessage call waits
 for a message to arrive in the queue before returning a response. If the number of
 messages in the queue is extremely small, you might not receive any messages in a
 particular ReceiveMessage response; in that case you should repeat the request.",
          "type": "number",
          "minimum": 0,
          "maximum": 20,
          "default": 0
        },
        "SQSVisibilityTimeout": {
          "description": "The number of seconds that the received messages are
 hidden from subsequent retrieve requests after being retrieved by a ReceiveMessage
 request.",
          "type": "number",
          "minimum": 0,
          "maximum": 43200
        }
      },
      "additionalProperties": false,
      "required": [
        "SQSQueueName"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1vd3y4ygbqmfk

Classifications:

- Management | Advanced stack components | Database Migration Service (DMS) | Stop replication task

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop DMS Replication Task",
  "description": "Stop a Database Migration Service (DMS) replication task. The
 specified task must be in the running state.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StopDmsTask.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StopDmsTask"
      ],
      "default": "AWSManagedServices-StopDmsTask"
    },
    "Region": {
      "description": "The AWS Region where the DMS Replication Task was created, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ReplicationTaskArn": {
          "description": "The DMS replication task Amazon resource name (ARN).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "arn:aws:dms:[a-z]{2}-[a-z]+-\\d{1}:\\d{12}:task:[A-Za-z0-9-]+
$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
```

```
        "ReplicationTaskArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1vjbacfr4ufdv

Classifications:

- Management | Advanced stack components | Security group | Revoke ingress rule

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Revoke Ingress Rule",
  "description": "Revoke the ingress rule for the specified security group (SG). You
 must specify the configurations of the ingress rule that you are revoking. Note that,
 once revoked, the ingress rule is permanently deleted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RevokeSecurityGroupIngressRuleV3.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RevokeSecurityGroupIngressRuleV3"
      ],
      "default": "AWSManagedServices-RevokeSecurityGroupIngressRuleV3"
    },
    "Region": {
```

```
      "description": "The AWS Region in which the security group is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityGroupId": {
          "description": "The ID of the security group (SG) that you are updating, in
 the form sg-0123456789abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IpProtocol": {
          "description": "The IP protocol name, or IP protocol number, for the ingress
 rule. For example, for TCP, enter either TCP, or (IP protocol number) 6. If you enter
 ICMP, you can specify any or all of the ICMP types and codes.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\+-\\\\(\\\\)\\w]{1,18}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "FromPort": {
          "description": "Start of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ToPort": {
```

```
          "description": "End of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Source": {
          "description": "An IP address range in CIDR notation, in the form
 255.255.255.255/32; or the ID of another security group in the same Region; or self,
 to specify the same security group.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8,17}$|^self$|^pl-\\w+|^[0-9]{12}\\/sg-[0-9a-f]{8,17}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SecurityGroupId",
          "IpProtocol",
          "FromPort",
          "ToPort",
          "Source"
        ]
      },
      "required": [
        "SecurityGroupId",
        "IpProtocol",
        "FromPort",
        "ToPort",
        "Source"
      ],
      "additionalProperties": false
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1vq0f289r36ay

Classifications:

- Management | Managed landing zone | Management account | Move account to OU

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Move Account To OU",
  "description": "Move an account under an AWS organizational unit (OU) to a different
 OU.",
  "type": "object",
  "properties": {
    "AccountId": {
      "description": "The unique identifier (ID) of the account that you want to
 move.",
      "type": "string",
      "pattern": "^[0-9]{12}$"
    },
    "TargetOUPath": {
      "description": "The path of the target OU that you want to move the account to.
 The path starts with either \"customer-managed\" or \"applications\". For example,
 \"applications:development\" and \"customer-managed:active\" are valid.",
      "type": "string",
      "pattern": "^([A-Za-z0-9-]+:[A-Za-z0-9-]+)+$|^[A-Za-z0-9-]+$"
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "AccountId",
      "TargetOUPath"
    ]
  },
  "additionalProperties": false,
  "required": [
    "AccountId",
    "TargetOUPath"
  ]
}
```

Schema for Change Type ct-1w8z66n899dct

Classifications:

- Management | AWS service | Self-provisioned service | Add

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Self-Provisioned AWS Service",
  "description": "Add a specific, allowed, AWS service to your AMS account. This
 CT validates prerequisites in the account and deploys a service with the default
 parameters. Not all Self-service provisioning services are supported, the ServiceName
 parameter for this CT lists the ones that are. For each service that you add, AMS
 creates a new role so you use the service without AMS management under the AMS Shared
 Responsibility model. Compliance is a shared responsibility and your AMS compliance
 status does not automatically apply to services or applications that you add in this
 way. Some AWS services do not have compliance certifications. For more information,
 see the AWS Services in Scope of AWS Assurance Program page. On that page, unless
 specifically excluded, features of each of the services are considered in scope of
 the assurance programs, and are reviewed and tested as part of our assessment when you
 submit this CT.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleCreateSSPSResources-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleCreateSSPSResources-Admin"
```

```
      ],
      "default": "AWSManagedServices-HandleCreateSSPSResources-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ServiceName": {
          "description": "The name of the AWS service.",
          "type": "string",
          "enum": [
            "AWS App Mesh",
            "AWS AppSync",
```

```
            "AWS Batch",
            "AWS Certificate Manager (ACM)",
            "AWS Private Certificate Authority (PCA)",
            "AWS CloudHSM",
            "AWS CodeBuild",
            "AWS CodeCommit",
            "AWS CodeDeploy",
            "AWS Device Farm",
            "AWS Elemental MediaStore",
            "AWS Elemental MediaTailor",
            "AWS Global Accelerator",
            "AWS Glue",
            "AWS License Manager",
            "AWS Migration Hub",
            "AWS Outposts",
            "AWS Resilience Hub",
            "AWS Security Hub",
            "AWS Service Catalog AppRegistry",
            "AWS Shield",
            "AWS Step Functions",
            "AWS Systems Manager Automation",
            "AWS Systems Manager Parameter Store",
            "AWS Transfer for SFTP",
            "AWS Transit Gateway",
            "AWS WAF - Web Application Firewall",
            "AWS X-Ray",
            "Amazon API Gateway",
            "Amazon Athena",
            "Amazon CloudSearch",
            "Amazon CloudWatch Synthetics",
            "Amazon Cognito",
            "Amazon DevOps Guru",
            "Amazon Directory Services - ADConnector Only",
            "Amazon DocumentDB (with MongoDB compatibility)",
            "Amazon DynamoDB",
            "Amazon ECR",
            "Amazon ECS on AWS Fargate",
            "Amazon EventBridge",
            "Amazon FSx",
            "Amazon FSx OnTap",
            "Amazon Forecast",
            "Amazon Inspector",
            "Amazon Kinesis Data Streams",
            "Amazon Kinesis Video Streams",
```

```
            "Amazon Lex",
            "Amazon Managed Service for Prometheus",
            "Amazon Managed Streaming for Apache Kafka",
            "Amazon MQ",
            "Amazon Pinpoint",
            "Amazon QuickSight",
            "Amazon SageMaker",
            "Amazon Simple Email Service",
            "Amazon Simple Workflow Service",
            "Amazon WorkDocs",
            "EC2 Image Builder"
          ]
        },
        "IAMRole": {
          "description": "An existing IAM console-access role name, or the Amazon
 resource name (ARN) of the role, to add the permissions to manage the AWS self-service
 provisioning service (SSPS). If left blank, a new role is created with the necessary
 permissions.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:role/[A-Za-z0-9_-]+$|
^[A-Za-z0-9_-]+$|^$"
        },
        "SAMLProviders": {
          "description": "A single SAML provider name or a comma-separated list of SAML
 providers to use with the role.",
          "type": "string",
          "pattern": "^[\\w+=,.@-]{0,256}$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "ServiceName",
          "IAMRole",
          "SAMLProviders"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ServiceName"
      ]
    }
  },
  "metadata": {
    "ui:order": [
```

```
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1wle0ai4en6km

Classifications:

- Management | Advanced stack components | EBS Volume | Modify

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Modify EBS Volumes",
  "description": "Modify EBS Volumes that are not attached to an EC2 instance in an
 Auto Scaling group. If you resize the volume, then you may need to extend the OS file
 system on the volume to use any newly allocated space. If a drift is introduced in
 the CloudFormation stack that was used to create the volume, then the automation can
 try to remediate the stack drift for stacks that are not created using CloudFormation
 ingest change type (ct-36cn2avfrrj9v).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ModifyEBSVolumes.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ModifyEBSVolumes"
      ],
      "default": "AWSManagedServices-ModifyEBSVolumes"
    },
    "Region": {
      "description": "The AWS Region where the EBS Volumes are located, in the form us-
east-1.",
      "type": "string",
```

```
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VolumeIds": {
          "description": "A list of up to 50 EBS volume IDs, in the form
vol-1234567890abcdef0.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^vol-([0-9a-f]{8}|[0-9a-f]{17})$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "CreateSnapshot": {
          "description": "True to create a snapshot before modifying the volume, False
to not. Default is True.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "VolumeType": {
          "description": "The desired volume type. If left unspecified, the existing
type is retained. Valid values are io1, io2, gp2, gp3, sc1, st1 and standard.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "io1",
              "io2",
              "gp2",
              "gp3",
              "sc1",
```

```
              "st1",
              "standard"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "VolumeSize": {
          "description": "The desired size of the volume, in GiB. The target volume
 size must be greater than or equal to the existing size of the volume. If left
 unspecified, the existing size is retained.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1-9]|[1-9][0-9]{1,3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4]
[0-9]{2}|655[0-2][0-9]|6553[0-6])$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Iops": {
          "description": "The requested number of I/O operations per second (IOPS).
 This parameter is only valid for io1, io2 and gp3 volumes. If left unspecified, the
 existing value is retained, unless the VolumeType is modified to one that supports
 different values. We highly recommend that you specify the desired Iops value when
 changing the VolumeType.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Throughput": {
          "description": "The throughput to provision for a volume, with a maximum
 of 1000 MiB/s. This parameter is valid only for gp3 volumes. If left unspecified, a
 minimum value is assigned or the existing value is retained.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused
by volume modification. False to not attempt drift remediation. Drift remediation can
be performed only on CloudFormation stacks that were created using a CT other than
the Ingestion CT ct-36cn2avfrrj9v and that are in sync with the definitions in the
stack template prior to the volume modification. Set to False to modify a volume in an
ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "VolumeIds",
          "CreateSnapshot",
          "VolumeType",
          "VolumeSize",
          "Iops",
          "Throughput",
          "RemediateStackDrift"
        ]
      },
      "required": [
        "VolumeIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
```

```
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1x66wvkjw2zp5

Classifications:

- Management | Advanced stack components | Target group | Update (for NLB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update target group for NLB",
  "description": "Use to update properties of an existing Target Group for a Network
 Load Balancer.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the Target Group (for NLB) that you are updating,
 in the form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "HealthCheckHealthyThreshold": {
          "type": "string",
```

```
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$"
        },
        "HealthCheckInterval": {
          "type": "integer",
          "description": "The approximate interval, in seconds, between health checks.
 Supported values are 10 or 30 seconds. Cannot change if the target protocol is TCP"
        },
        "HealthCheckTargetPath": {
          "type": "string",
          "description": "The ping path destination on the application hosts
 where the load balancer sends health check requests. Only applicable if
 HealthCheckTargetProtocol = HTTP or HTTPS."
        },
        "HealthCheckTargetPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health checks
 on targets. The default is traffic-port, which indicates the port on which each target
 receives traffic from the load balancer.",
          "pattern": "[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2]
[0-9]|6553[0-5]|traffic-port|"
        },
        "HealthCheckTargetProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS",
            "TCP"
          ]
        },
        "ProxyProtocolV2": {
          "type": "string",
          "description": "True if proxy protocol version 2 is enabled. False if it is
 not.",
          "enum": [
            "true",
            "false"
          ]
        },
        "DeregistrationDelayTimeout": {
          "type": "string",
```

```
          "description": "The amount of time, in seconds, for Elastic Load Balancing to
 wait before changing the state of a deregistering target from draining to unused.",
          "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})"
        },
        "Target1ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target1Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target1AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target1ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target1ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target1ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        },
        "Target2ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
```

```
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target2ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target2ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target2ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        },
        "Target3ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target3AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target3ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target3ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target3ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|$"
        },
        "Target4ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
```

```
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target4AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target4ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target4ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target4ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        },
        "Target5ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target5Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target5AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target5ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target5ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
```

```
 IP address in Target5ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        },
        "Target6ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target6Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target6AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target6ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target6ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target6ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        },
        "Target7ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target7Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
```

```
        },
        "Target7AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target7ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target7ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target7ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|$"
        },
        "Target8ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target8Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target8AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target8ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target8ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target8ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "DeregistrationDelayTimeout",
          "ProxyProtocolV2",
          "HealthCheckTargetPath",
```

```
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckInterval",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
          "Target4AvailabilityZone",
          "Target5ID",
          "Target5Port",
          "Target5AvailabilityZone",
          "Target6ID",
          "Target6Port",
          "Target6AvailabilityZone",
          "Target7ID",
          "Target7Port",
          "Target7AvailabilityZone",
          "Target8ID",
          "Target8Port",
          "Target8AvailabilityZone"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
```

```
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-1yq7hhqse71yg

Classifications:

- Management | Advanced stack components | Database Migration Service (DMS) | Start replication task

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start DMS Replication Task",
  "description": "Start a new Database Migration Service (DMS) replication task, or a
 task in a stopped or failed state.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartDmsTask.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartDmsTask"
      ],
      "default": "AWSManagedServices-StartDmsTask"
    },
    "Region": {
      "description": "The AWS Region where the DMS replication task was created, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ReplicationTaskArn": {
          "description": "The DMS replication task Amazon resource name (ARN).",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "arn:aws:dms:[a-z]{2}-[a-z]+-\\d{1}:\\d{12}:task:[A-Za-z0-9-]+
$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "StartReplicationTaskType": {
          "description": "The type of DMS replication task. To start a new task, use
 start-replication. To restart a stopped task or failed task from the CDC position
 where the task stopped, use resume-processing. To restart a stopped or failed task of
 type full-load or full-load-and-cdc, use reload-target.",
          "type": "array",
          "items": {
            "enum": [
              "start-replication",
              "resume-processing",
              "reload-target"
            ],
            "type": "string",
            "default": "start-replication"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CdcStartPosition": {
          "description": "When to start the change data capture (CDC) operation. Use a
 timestamp in the format (yyyy-mm-ddThh:mm:ss), a log sequence number, or a checkpoint
 (either source database-engine specific, or AWS DMS-specific).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^\\d{1,4}-\\d{2}-\\d{2}T\\d{2}:\\d{2}:\\d{2}$|^checkpoint:\
\w{1}\\d{1}\\#\\d{2}\\#[a-z]+-[a-z]+-[a-z]+.[0-9]+:[0-9]+:[-0-9]+:[0-9]+:[0-9]+:[a-z]+-
[a-z]+-[a-z]+.[0-9]+:[0-9]+\\#\\d{1}\\#\\d{1}\\#\\*\\#\\d{1}\\#\\d{2}$|^[a-z]+-[a-z]+-
[a-z]+.[0-9]+:[0-9]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CdcStopPosition": {
          "description": "The timestamp in the format (server_time:yyyy-mm-ddThh:mm:ss)
 to stop the change data capture (CDC) operation.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^$|^server_time:\\d{1,4}-\\d{2}-\\d{2}T\\d{2}:\\d{2}:\\d{2}$|
^commit_time:[\\s]?\\d{1,4}-\\d{2}-\\d{2}T\\d{2}:\\d{2}:\\d{2}[\\s]?$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ReplicationTaskArn",
        "StartReplicationTaskType"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-1yqy4frl5s8y8

Classifications:

- Management | Managed landing zone | Management account | Delete StackSets stack (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete StackSets Stack",
  "description": "Delete AWS CloudFormation (CFN) StackSets-created stacks and
 instances.",
  "type": "object",
  "properties": {
    "Name": {
      "description": "Name of the StackSets stack to be deleted.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$",
      "maxLength": 128
    },
    "Region": {
      "description": "The AWS Region to delete the resources, in the form of us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Region",
      "Priority"
    ]
  },
  "required": [
    "Name"
  ]
}
```

Schema for Change Type ct-1zdasmc2ewzrs

Classifications:

- Deployment | Managed landing zone | Management account | Create application account (with VPC)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Application Account With VPC",
  "description": "Create a managed AWS landing zone application account and a VPC with
 up to 10 private subnets and up to 5 optional public subnets per availability zone
 (AZ) for two or three AZ's. Optionally, also create an AWS Backup plan with up to four
 different rules. Managed AWS landing zone core accounts must already be onboarded to
 AWS Managed Services (AMS).",
  "type": "object",
  "properties": {
    "AccountName": {
      "description": "A name for the new application account. Max length 50 characters.
 The underscore (_) is not allowed.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9]{1}[a-zA-Z0-9.-]{0,49}$"
    },
    "AccountEmail": {
      "description": "The email address for the new application account. The email must
 be unique per application account.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
    },
    "ApplicationOUName": {
      "description": "The name of an existing organizational unit (OU) for this
 application account, in the form of <application ou name>:<child ou name>. The default
 value is applications:managed.",
      "type": "string",
      "default": "applications:managed"
    },
    "SupportLevel": {
      "description": "The account's AMS support level, Premium or Plus.",
      "type": "string",
      "enum": [
        "plus",
        "premium"
```

```
      ]
    },
    "VpcName": {
      "description": "A meaningful name for the application account VPC. Must be unique
 within this application account.",
      "type": "string"
    },
    "NumberOfAZs": {
      "description": "The number of availability zones (AZs) that the VPC supports.
 Options are 2 or 3.",
      "type": "number",
      "minimum": 2,
      "maximum": 3
    },
    "VpcCIDR": {
      "description": "The Classless Inter-Domain Routing (CIDR) for the VPC.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "RouteType": {
      "description": "The AWS Transit Gateway application route table connection type.
 For this VPC to accept connections from other VPCs, use routable. For it to not accept
 those connections, use isolated. The default is routable.",
      "type": "string",
      "enum": [
        "isolated",
        "routable"
      ],
      "default": "routable"
    },
    "TransitGatewayApplicationRouteTableName": {
      "description": "The existing AWS Transit Gateway route table for this application
 account VPC. The default is defaultAppRouteDomain. To create a new application route
 table, use the Create Application Route Table change type.",
      "type": "string",
      "default": "defaultAppRouteDomain"
    },
    "PublicSubnetAZ1CIDR": {
      "description": "The CIDR for the optional first public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
```

```
    },
    "PublicSubnetAZ2CIDR": {
      "description": "The CIDR for the optional first public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnetAZ3CIDR": {
      "description": "The CIDR for the optional first public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ1CIDR": {
      "description": "The CIDR for the optional second public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ2CIDR": {
      "description": "The CIDR for the optional second public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ3CIDR": {
      "description": "The CIDR for the optional second public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ1CIDR": {
      "description": "The CIDR for the optional third public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ2CIDR": {
```

```
      "description": "The CIDR for the optional third public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ3CIDR": {
      "description": "The CIDR for the optional third public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ1CIDR": {
      "description": "The CIDR for the optional fourth public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ2CIDR": {
      "description": "The CIDR for the optional fourth public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ3CIDR": {
      "description": "The CIDR for the optional fourth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ1CIDR": {
      "description": "The CIDR for the optional fifth public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ2CIDR": {
      "description": "The CIDR for the optional fifth public subnet in availability
 zone 2.",
```

```
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ3CIDR": {
      "description": "The CIDR for the optional fifth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ1CIDR": {
      "description": "The CIDR for the first private subnet in availability zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ2CIDR": {
      "description": "The CIDR for the first private subnet in availability zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ3CIDR": {
      "description": "The CIDR for the first private subnet in optional availability
 zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet2AZ1CIDR": {
      "description": "The CIDR for the optional second private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet2AZ2CIDR": {
      "description": "The CIDR for the optional second private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
```

```
    "PrivateSubnet2AZ3CIDR": {
      "description": "The CIDR for the optional second private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ1CIDR": {
      "description": "The CIDR for the optional third private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ2CIDR": {
      "description": "The CIDR for the optional third private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ3CIDR": {
      "description": "The CIDR for the optional third private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ1CIDR": {
      "description": "The CIDR for the optional fourth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ2CIDR": {
      "description": "The CIDR for the optional fourth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ3CIDR": {
```

```
      "description": "The CIDR for the optional fourth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ1CIDR": {
      "description": "The CIDR for the optional fifth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ2CIDR": {
      "description": "The CIDR for the optional fifth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ3CIDR": {
      "description": "The CIDR for the optional fifth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ1CIDR": {
      "description": "The CIDR for the optional sixth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ2CIDR": {
      "description": "The CIDR for the optional sixth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ3CIDR": {
      "description": "The CIDR for the optional sixth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
```

```
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ1CIDR": {
      "description": "The CIDR for the optional seventh private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ2CIDR": {
      "description": "The CIDR for the optional seventh private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ3CIDR": {
      "description": "The CIDR for the optional seventh private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ1CIDR": {
      "description": "The CIDR for the optional eighth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ2CIDR": {
      "description": "The CIDR for the optional eighth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ3CIDR": {
      "description": "The CIDR for the optional eighth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
```

```
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ1CIDR": {
      "description": "The CIDR for the optional ninth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ2CIDR": {
      "description": "The CIDR for the optional ninth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ3CIDR": {
      "description": "The CIDR for the optional ninth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet10AZ1CIDR": {
      "description": "The CIDR for the optional tenth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet10AZ2CIDR": {
      "description": "The CIDR for the optional tenth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet10AZ3CIDR": {
      "description": "The CIDR for the optional tenth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
```

```
    },
    "DirectAlertsEmail": {
      "description": "Email address to receive specifically tagged resource-based
 alerts, and the onboarding process will create your SNS subscription. If not
 specified, then you can subscribe later using the DirectCustomerAlerts change type
 (ct-t-3rcl9u1k017wu).",
      "type": "string",
      "pattern": "^[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-
zA-Z0-9])?(?:\\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)*$"
    },
    "SamlMetadataDocumentURL": {
      "description": "The URL that points to the Security Assertion Markup
 Language(SAML) metadata document that is used to enable federated access to the
 application account. Typically, a pre-signed URL for an Amazon S3 object.",
      "type": "string",
      "pattern": "^https://.+$|^$|s3://.+$"
    },
    "BackupPlanName": {
      "type": "string",
      "description": "A meaningful name for the AWS Backup plan, which is a policy
 expression that defines when and how you want to back up your AWS resources.",
      "default": "default-backup-plan"
    },
    "ResourceTagKey": {
      "type": "string",
      "description": "The tag key (case sensitive) of the resources to be backed up.
 For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
      "default": "Backup"
    },
    "ResourceTagValue": {
      "type": "string",
      "description": "The tag value (case sensitive) of the resources to be backed up.
 For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
      "default": "True"
    },
    "BackupRule1ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
 initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
 UTC time.",
      "type": "string",
```

```
      "pattern": "^(cron|rate)\\(.*\\)$",
      "default": "cron(0 2 ? * * )"
    },
    "BackupRule1DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the daily backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 7
    },
    "BackupRule1MoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the daily backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule2ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$"
    },
    "BackupRule2DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that weekly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule2MoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that weekly backups are moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600,
```

```
      "default": 0
    },
    "BackupRule3ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$"
    },
    "BackupRule3DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that monthly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule3MoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the monthly backups are
moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
the backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule4ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$"
    },
    "BackupRule4DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the yearly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule4MoveToColdStorageAfterDays": {
```

```
      "type": "integer",
      "description": "The number of days after creation that the yearly backups are
moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
the backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "PatchOrchestratorFirstTagKey": {
      "description": "The first tag-key to use for creating your \"Patch Group\" tag
values. For example, AppId. Specify null if you already have defined your own patch
groups with a \"Patch Group\" tag.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$"
    },
    "PatchOrchestratorSecondTagKey": {
      "description": "The second tag-key to use for creating your \"Patch Group\" tag
values. For example, Environment. Specify null if you already have defined your own
patch groups with a \"Patch Group\" tag.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$"
    },
    "PatchOrchestratorThirdTagKey": {
      "description": "The third tag-key to use for creating your \"Patch Group\" tag
values. For example, Group. Specify null if you already have defined your own patch
groups with a \"Patch Group\" tag.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$"
    },
    "PatchOrchestratorDefaultMaintenanceWindowCutoff": {
      "description": "The number of hours before the end of the Default Maintenance
Window in which no new patching commands are started. This interval exists to allow
enough time for patching to complete before the window ends.",
      "minimum": 0,
      "maximum": 23,
      "type": "integer"
    },
    "PatchOrchestratorDefaultMaintenanceWindowDuration": {
      "description": "The duration of the maintenance window in hours.",
      "minimum": 1,
      "maximum": 24,
      "type": "integer"
    },
    "PatchOrchestratorDefaultMaintenanceWindowSchedule": {
```

```
      "description": "The schedule of the maintenance window in the form of a cron or
rate expression. For example cron(0 18 * * ? *) would create a window at 18:00 every
day, and rate(7 days) would create a window every seven days.",
      "minLength": 1,
      "maxLength": 256,
      "pattern": "^cron\\([0-9a-zA-Z\\ ?*#-,\\/]+\\)$|^rate\\([0-9a-zA-Z\\ ]+\\)$",
      "type": "string"
    },
    "PatchOrchestratorDefaultMaintenanceWindowTimeZone": {
      "description": "The time zone that the scheduled maintenance window executions
are based on, in Internet Assigned Numbers Authority (IANA) format.",
      "pattern": "^[a-zA-Z_]+(\\+|/)?[a-zA-Z0-9_-]*(\\+|/)?[a-zA-Z0-9_-]+$",
      "type": "string"
    },
    "PatchOrchestratorDefaultPatchBackupRetentionInDays": {
      "description": "The number of days the backup taken before patching will remain
available.",
      "minimum": 1,
      "maximum": 90,
      "type": "integer"
    },
    "PatchOrchestratorNotificationEmails": {
      "description": "One or more email addresses to receive notifications about
default patching status. Use group distribution lists instead of individual emails.",
      "items": {
        "type": "string",
        "pattern": "^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$"
      },
      "minItems": 1,
      "maxItems": 5,
      "type": "array",
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "AccountName",
      "AccountEmail",
      "ApplicationOUName",
      "SupportLevel",
      "DirectAlertsEmail",
      "SamlMetadataDocumentURL",
      "VpcName",
      "VpcCIDR",
```

```
        "NumberOfAZs",
        "RouteType",
        "TransitGatewayApplicationRouteTableName",
        "PublicSubnetAZ1CIDR",
        "PublicSubnetAZ2CIDR",
        "PublicSubnetAZ3CIDR",
        "PublicSubnet2AZ1CIDR",
        "PublicSubnet2AZ2CIDR",
        "PublicSubnet2AZ3CIDR",
        "PublicSubnet3AZ1CIDR",
        "PublicSubnet3AZ2CIDR",
        "PublicSubnet3AZ3CIDR",
        "PublicSubnet4AZ1CIDR",
        "PublicSubnet4AZ2CIDR",
        "PublicSubnet4AZ3CIDR",
        "PublicSubnet5AZ1CIDR",
        "PublicSubnet5AZ2CIDR",
        "PublicSubnet5AZ3CIDR",
        "PrivateSubnet1AZ1CIDR",
        "PrivateSubnet1AZ2CIDR",
        "PrivateSubnet1AZ3CIDR",
        "PrivateSubnet2AZ1CIDR",
        "PrivateSubnet2AZ2CIDR",
        "PrivateSubnet2AZ3CIDR",
        "PrivateSubnet3AZ1CIDR",
        "PrivateSubnet3AZ2CIDR",
        "PrivateSubnet3AZ3CIDR",
        "PrivateSubnet4AZ1CIDR",
        "PrivateSubnet4AZ2CIDR",
        "PrivateSubnet4AZ3CIDR",
        "PrivateSubnet5AZ1CIDR",
        "PrivateSubnet5AZ2CIDR",
        "PrivateSubnet5AZ3CIDR",
        "PrivateSubnet6AZ1CIDR",
        "PrivateSubnet6AZ2CIDR",
        "PrivateSubnet6AZ3CIDR",
        "PrivateSubnet7AZ1CIDR",
        "PrivateSubnet7AZ2CIDR",
        "PrivateSubnet7AZ3CIDR",
        "PrivateSubnet8AZ1CIDR",
        "PrivateSubnet8AZ2CIDR",
        "PrivateSubnet8AZ3CIDR",
        "PrivateSubnet9AZ1CIDR",
        "PrivateSubnet9AZ2CIDR",
```

```
      "PrivateSubnet9AZ3CIDR",
      "PrivateSubnet10AZ1CIDR",
      "PrivateSubnet10AZ2CIDR",
      "PrivateSubnet10AZ3CIDR",
      "BackupPlanName",
      "ResourceTagKey",
      "ResourceTagValue",
      "BackupRule1ScheduleExpression",
      "BackupRule1DeleteAfterDays",
      "BackupRule1MoveToColdStorageAfterDays",
      "BackupRule2ScheduleExpression",
      "BackupRule2DeleteAfterDays",
      "BackupRule2MoveToColdStorageAfterDays",
      "BackupRule3ScheduleExpression",
      "BackupRule3DeleteAfterDays",
      "BackupRule3MoveToColdStorageAfterDays",
      "BackupRule4ScheduleExpression",
      "BackupRule4DeleteAfterDays",
      "BackupRule4MoveToColdStorageAfterDays",
      "PatchOrchestratorFirstTagKey",
      "PatchOrchestratorSecondTagKey",
      "PatchOrchestratorThirdTagKey",
      "PatchOrchestratorDefaultMaintenanceWindowCutoff",
      "PatchOrchestratorDefaultMaintenanceWindowDuration",
      "PatchOrchestratorDefaultMaintenanceWindowSchedule",
      "PatchOrchestratorDefaultMaintenanceWindowTimeZone",
      "PatchOrchestratorDefaultPatchBackupRetentionInDays",
      "PatchOrchestratorNotificationEmails"
    ]
  },
  "additionalProperties": false,
  "required": [
    "AccountName",
    "AccountEmail",
    "SupportLevel",
    "VpcName",
    "VpcCIDR",
    "NumberOfAZs",
    "PrivateSubnet1AZ1CIDR",
    "PrivateSubnet1AZ2CIDR",
    "BackupPlanName",
    "ResourceTagKey",
    "ResourceTagValue",
    "BackupRule1ScheduleExpression"
```

```
  ]
}
```

Schema for Change Type ct-2019s9y3nfml4

Classifications:

- Management | Directory Service | Users and groups | Remove user from group

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove AD User From AD Group",
  "description": "Remove an Active Directory (AD) user from an AD group in the AMS
 managed AD. For multi-account landing zone (MALZ), use this change type in the shared
 services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RemoveADUserFromGroup-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveADUserFromGroup-Admin"
      ],
      "default": "AWSManagedServices-RemoveADUserFromGroup-Admin"
    },
    "Region": {
      "description": "The AWS Region where the AMS managed AD is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "UserName": {
          "description": "The name of the AD user.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|\
\@]{2,19}[^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|\\@\\.]$"
          },
```

```
          "maxItems": 1,
          "minItems": 1
        },
        "GroupName": {
          "description": "The name of the AD group to remove the user from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|]
[^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|]{0,61}[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|]$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "DomainFQDN": {
          "description": "The fully qualified domain name (FQDN) where the user exists,
 this can be the AMS managed or trusted domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "UserName",
          "GroupName",
          "DomainFQDN"
        ]
      },
      "required": [
        "UserName",
        "GroupName",
        "DomainFQDN"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
```

```
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2052miu12d8fn

Classifications:

- Management | Advanced stack components | RDS database stack | Update master user password

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update RDS MasterUserPassword",
  "description": "Update the MasterUserPassword property of an Amazon Relational
 Database Service (RDS) database instance.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateInstanceMasterUserPasswordV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateInstanceMasterUserPasswordV2"
      ],
      "default": "AWSManagedServices-UpdateInstanceMasterUserPasswordV2"
    },
    "Region": {
      "description": "The AWS Region of the account with the RDS database instance; for
 example, us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
      "DBInstanceIdentifier": {
        "description": "The identifier of the RDS database instance; for example,
mydbinstance.",
        "type": "string",
        "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
      },
      "SecretName": {
        "description": "The name of the Secrets Manager secret that stores the new
RDS master user password, You must specify either this property, or \"SSMParameter\",
but not both.",
        "type": "string",
        "pattern": "^$|^[a-zA-Z0-9\\_\\.\\-\\/\\=\\@]{0,255}$",
        "default": ""
      },
      "SecretKey": {
        "description": "The \"Key\" in the Secrets Manager secret that stores the new
RDS master user password, required only if SecretName is provided.",
        "type": "string",
        "pattern": "^$|^[a-zA-Z0-9\\_\\.\\-\\/\\=\\@]{0,255}$",
        "default": ""
      },
      "SSMParameter": {
        "description": "The name of the SSM Parameter Store parameter that stores new
RDS master user password. You must specify either this property, or \"SecretName\",
but not both.",
        "type": "string",
        "pattern": "^$|^[a-zA-Z0-9\\_\\.\\-]{0,255}$",
        "default": ""
      }
    },
    "metadata": {
      "ui:order": [
        "DBInstanceIdentifier",
        "SecretName",
        "SecretKey",
        "SSMParameter"
      ]
    },
    "additionalProperties": false,
    "required": [
      "DBInstanceIdentifier"
    ]
  }
 },
```

```
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-20san5sgtwd9e

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create from snapshot

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS Instance From Snapshot",
  "description": "Create an Amazon Relational Database Service (RDS) DB instance from
 an RDS snapshot.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "The ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-siqajx20000000000.",
```

```
      "type": "string",
      "enum": [
        "stm-siqajx20000000000"
      ]
    },
    "Name": {
      "description": "A name for the stack; this becomes the Stack Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
```

```
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 720
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "DBInstanceClass": {
          "description": "The compute and memory capacity for the DB instance. To
inherit this value from the snapshot, use inherit.",
          "type": "string",
          "pattern": "^inherit$|^db\\.[a-z0-9]+\\.[a-z0-9]+$",
          "default": "inherit"
        },
        "DBInstanceIdentifier": {
          "description": "A name for the DB instance. If you specify a name, it is
converted to lowercase. If you don't specify a name, a unique physical ID is generated
and used for the DBInstanceIdentifier.",
          "type": "string",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$",
          "default": ""
        },
        "DBSnapshotIdentifier": {
          "description": "The name of the RDS DB snapshot to use to create the DB
instance.",
          "type": "string"
        },
        "DBDomain": {
          "description": "The directory ID of the Active Directory to create the
instance in. To use DBDomain, you must provide an eligible SQL Server, Oracle, or
Postgres engine in the DBEngine field.",
          "type": "string",
          "pattern": "^$|^d-[0-9a-f]{10}$"
        },
        "DBDomainIAMRoleName": {
          "description": "The name of an IAM role that Amazon RDS uses when calling the
AWS Directory Service APIs.",
```

```
          "type": "string",
          "pattern": "^$|^customer[\\w-]+$"
        },
        "DBEngine": {
          "description": "The name of the database engine for the DB instance. Must
be compatible with the engine of the source. If not specified, it will default to
the same engine as the source. Not every database engine is available for every AWS
region.",
          "type": "string"
        },
        "DBOptionGroupName": {
          "description": "The option group that this DB instance is associated with.
If none is provided, the default option group is associated. An option group can
specify features, called options, that are available for a particular Amazon RDS DB
instance.",
          "type": "string"
        },
        "DBParameterGroupName": {
          "description": "The name of an existing DB parameter group. If none is
provided, the default parameter group is associated. A DB parameter group acts
as a container for engine configuration values that are applied to one or more DB
instances.",
          "type": "string"
        },
        "DBSubnetIds": {
          "description": "Two or more subnet IDs for the DB instance, in the form
subnet-0123abcd or subnet-01234567890abcdef, spanning at least two Availability
Zones.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 2,
          "maxItems": 20,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "DBInstanceClass",
          "DBInstanceIdentifier",
          "DBSnapshotIdentifier",
```

```
            "DBDomain",
            "DBDomainIAMRoleName",
            "DBEngine",
            "DBOptionGroupName",
            "DBParameterGroupName",
            "DBSubnetIds"
          ]
        },
        "required": [
          "DBSnapshotIdentifier",
          "DBSubnetIds"
        ]
      }
    },
    "additionalProperties": false,
    "metadata": {
      "ui:order": [
        "Name",
        "Description",
        "VpcId",
        "Parameters",
        "TimeoutInMinutes",
        "StackTemplateId",
        "Tags"
      ]
    },
    "required": [
      "Description",
      "VpcId",
      "StackTemplateId",
      "Name",
      "TimeoutInMinutes",
      "Parameters"
    ]
}
```

Schema for Change Type ct-211l2gxvsrrhy

Classifications:

- Management | Advanced stack components | EC2 instance stack | Enable detailed monitoring (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Detailed Monitoring",
  "description": "Enable detailed monitoring for the specified EC2 instance. Detailed
 monitoring incurs a charge. EC2 detailed monitoring provides more frequent metrics,
 published at one-minute intervals, instead of the five-minute intervals used in Amazon
 EC2 basic monitoring.",
  "type": "object",
  "properties": {
    "InstanceIds": {
      "description": "A list of up to 20 EC2 instance IDs, in the form
 i-1234567890abcdef0 or i-b188560f.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
      },
      "minItems": 1,
      "maxItems": 20,
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "InstanceIds",
      "Priority"
    ]
  },
  "required": [
    "InstanceIds"
  ]
}
```

Schema for Change Type ct-220bdb8blaixf

Classifications:

- Deployment | Advanced stack components | S3 storage | Create policy (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create policy",
  "description": "Create an S3 bucket policy. The existing bucket policy (if any) is
 replaced with the new policy.",
  "type": "object",
  "properties": {
    "BucketName": {
      "description": "The name of the Amazon S3 bucket to which the policy applies.",
      "type": "string",
      "pattern": "^[A-Za-z0-9][A-Za-z0-9\\-]{1,61}[A-Za-z0-9]$",
      "maxLength": 63
    },
    "BucketPolicy": {
      "description": "Detailed information about the bucket permissions, or a policy
 document to be attached to the bucket (paste the policy document into the value
 field). Details should include the type of access (for example Read, Write, or
 Delete). If it is a valid policy document, it replaces the existing bucket policy.
 If you want to append a new statement or modify an existing statement on the bucket
 policy, paste in the complete bucket policy with the new or modified statements.",
      "type": "string",
      "maxLength": 20000
    },
    "Operation": {
      "description": "Must be Create policy.",
      "type": "string",
      "default": "Create policy",
      "enum": [
        "Create policy"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
```

```
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "BucketName",
      "BucketPolicy",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "BucketName",
    "BucketPolicy",
    "Operation"
  ]
}
```

Schema for Change Type ct-22cbvc1yujhec

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Reset service-specific credentials

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Reset Service-Specific Credentials",
  "description": "Reset the password for the specified service-specific credential.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ResetServiceSpecificCredentials.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ResetServiceSpecificCredentials"
      ],
```

```
      "default": "AWSManagedServices-ResetServiceSpecificCredentials"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Username": {
          "description": "The name of the IAM user associated with the service-specific
 credential.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[\\w+=,.@-]+"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ServiceSpecificCredentialId": {
          "description": "The unique identifier for the service-specific credential.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[\\w]+"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SecretArn": {
          "description": "The ARN of the Secrets Manager secret that stores the
 credentials currently.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):secretsmanager:[a-z0-9-]+:[0-9]
{12}:secret:[a-zA-Z0-9-@.+=_/]{1,512}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "Username",
          "ServiceSpecificCredentialId",
          "SecretArn"
        ]
      },
      "required": [
        "Username",
        "ServiceSpecificCredentialId",
        "SecretArn"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-24pi85mjtza8k

Classifications:

- Management | Directory Service | Users and groups | Add user to group

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add AD User To AD Group",
```

```
  "description": "Add an Active Directory (AD) user to an AD group in the AMS managed
 AD. For multi-account landing zone (MALZ), use this change type in the shared services
 account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddADUserToGroup-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddADUserToGroup-Admin"
      ],
      "default": "AWSManagedServices-AddADUserToGroup-Admin"
    },
    "Region": {
      "description": "The AWS Region where the AMS managed AD is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "UserName": {
          "description": "The name of the AD user, do not include the domain name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|\
\@]{2,19}[^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\\|\\@\\.]$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "GroupName": {
          "description": "The name of the AD group to which the user is added. The
 group must exist in AMS managed AD and must belong to the CustomerGroups OU. The group
 scope must be DomainLocal.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|
\\\\][^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|\\\\]{0,61}[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\
\|]$"
          },
```

```
          "maxItems": 1,
          "minItems": 1
        },
        "DomainFQDN": {
          "description": "The fully qualified domain name (FQDN) where the user exists,
this can be the AMS managed or trusted domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "UserName",
          "GroupName",
          "DomainFQDN"
        ]
      },
      "required": [
        "UserName",
        "GroupName",
        "DomainFQDN"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
```

```
}
```

Schema for Change Type ct-257p9zjk14ija

Classifications:

- Deployment | Ingestion | Stack from migration partner migrated instance | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Migrate Instance to AMS Stack",
  "description": "Migrate a running non-AMS instance into an AMS stack, in a given
 AMS-managed VPC and subnet. Must be an instance that was configured through a cloud
 migration service. Tags that exist on the instance to be migrated will be applied to
 the resources created in addition to tags requested in the RFC. Number of total tags
 between the instance to be migrated and the resources created cannot exceed fifty. Set
 a Name tag to give the EC2 instance, and AMI, names in the EC2 console. Please note
 that your RFC will be rejected if a tag on the instance to be migrated has the same
 key as a tag supplied in the RFC.",
  "type": "object",
  "properties": {
    "InstanceId": {
      "description": "ID of a running instance to migrate, in the form i-0123abcd or
 i-01234567890abcdef.",
      "type": "string",
      "pattern": "^i-[a-zA-Z0-9]{8}$|^i-[a-zA-Z0-9]{17}$"
    },
    "TargetVpcId": {
      "description": "ID of the existing AMS VPC to deploy the migrated stack into, in
 the form vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "TargetSubnetId": {
      "description": "ID of the existing AMS subnet to deploy the migrated stack into,
 in the form subnet-0123abcd or subnet-01234567890abcdef.",
      "type": "string",
      "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
    },
    "TargetSecurityGroupIds": {
```

```
      "description": "IDs of the existing security groups to associate with the
migrated stack, in the form sg-0123abcd or sg-01234567890abcdef. If nothing is
specified, the default AMS security groups will be applied.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "TargetInstanceType": {
      "description": "The type of EC2 instance to deploy from the migrated instance.",
      "type": "string",
      "default": "t2.large"
    },
    "ApplyInstanceValidation": {
      "description": "True to run AMS pre-migration validation checks on the instance.
False to not run the checks. Default is true.",
      "type": "boolean",
      "default": true
    },
    "KmsKeyId": {
      "description": "KMS key to automatically encrypt the resulting AMI with. Use any
format specified in the AWS EC2 CopyImage API documentation.",
      "type": "string",
      "metadata": {
        "ams:sensitive": true
      }
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "EnforceIMDSV2": {
```

```
      "description": "Set to 'false' for the instance to be launched with IMDSv1 only.
Default value is 'true'. See EC2/IMDS document for more details.",
      "type": "string",
      "enum": [
        "true",
        "false"
      ],
      "default": "true"
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resources
created (AMI and EC2 instance). Set a Name tag to give the EC2 instance, and AMI,
names in the EC2 console.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "InstanceId",
      "TargetVpcId",
      "TargetSubnetId",
      "TargetSecurityGroupIds",
      "TargetInstanceType",
      "ApplyInstanceValidation",
      "KmsKeyId",
      "Name",
      "Description",
      "EnforceIMDSV2",
      "Tags"
    ]
  },
  "additionalProperties": false,
  "required": [
    "InstanceId",
    "TargetVpcId",
    "TargetSubnetId",
    "Name",
    "Description",
    "EnforceIMDSV2"
  ]
}
```

Schema for Change Type ct-25v6r7t8gvkq5

Classifications:

- Deployment | Monitoring and notification | GuardDuty threat intel set | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create GuardDuty ThreatIntelSet",
  "description": "Use to create an Amazon GuardDuty ThreatIntelSet instance, which is a
 list of known malicious IP addresses that have been blacklisted for communication with
 your AWS environment.",
  "type": "object",
  "properties": {
    "Activate": {
```

```
      "description": "Specified whether the ThreatIntelSet is active or not.",
      "type": "boolean",
      "default": true
    },
    "DetectorId": {
      "description": "The detector ID that specifies the GuardDuty service to which
you want to add a ThreatIntelSet. Leave this blank to use the only detector in the
selected region (this will not succeed if there is more than one detector in the
selected region).",
      "pattern": "^[a-fA-F0-9]{32}$|^$",
      "type": "string"
    },
    "Format": {
      "default": "TXT",
      "description": "The format of the file that contains the ThreatIntelSet.",
      "enum": [
        "TXT",
        "STIX",
        "OTX_CSV",
        "ALIEN_VAULT",
        "PROOF_POINT",
        "FIRE_EYE"
      ],
      "type": "string"
    },
    "Name": {
      "description": "The friendly name to identify the ThreatIntelSet. This name is
displayed in all findings that are triggered by activity that involves IP addresses
included in this ThreatIntelSet.",
      "minLength": 1,
      "type": "string"
    },
    "ThreatIntelSet": {
      "description": "The URI of the file that contains the ThreatIntelSet.",
      "minLength": 1,
      "type": "string"
    },
    "Region": {
      "description": "The region containing the GuardDuty detector to use; in the form
of us-east-1.",
      "minLength": 1,
      "type": "string"
    },
    "Priority": {
```

```
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Name",
      "ThreatIntelSet",
      "Format",
      "Activate",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Name",
    "ThreatIntelSet",
    "Region"
  ]
}
```

Schema for Change Type ct-26vhhlj9jmlpf

Classifications:

- Management | Advanced stack components | AMI | Deregister

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Deregister AMIs",
  "description": "Deregister one or multiple Amazon Machine Images (AMI)s and
 optionally delete all associated snapshots. Once deregistered the AMI or AMIs can't be
 used for launching new instances.",
  "type": "object",
```

```
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-BulkDeleteOrDeregisterAMI.",
      "type": "string",
      "enum": [
        "AWSManagedServices-BulkDeleteOrDeregisterAMI"
      ],
      "default": "AWSManagedServices-BulkDeleteOrDeregisterAMI"
    },
    "Region": {
      "description": "The AWS Region where the AMI or AMIs are located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ImageIds": {
          "description": "A comma-delimited list of up to 50 Amazon Machine Image IDs,
 in the form ami-0123abcd or ami-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^ami-[a-f0-9]{8,17}$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "DeleteSnapshots": {
          "description": "True (lower case) to delete all associated snapshots, false
 to not. The deletion of snapshots cannot be rolled back. Default is false.",
          "type": "array",
          "items": {
            "type": "boolean",
            "default": false
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ImageIds",
```

```
          "DeleteSnapshots"
        ]
      },
      "required": [
        "ImageIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2781aqd6f6svs

Classifications:

- Management | Advanced stack components | EC2 instance stack | Change hostname (Linux)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change Linux Hostname",
  "description": "Change the hostname of an EC2 Linux instance. If no hostname is
 provided, then the hostname is randomized.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ChangeHostname.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ChangeHostname"
```

```
      ],
      "default": "AWSManagedServices-ChangeHostname"
    },
    "Region": {
      "description": "The AWS Region where the EC2 instance is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance, in the form
 i-1234567890abcdef0.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Hostname": {
          "description": "A new hostname for the instance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9-]{1,63}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Platform": {
          "description": "Must be linux. To change the hostname for a Windows instance,
 use CT ct-0h3p576mj4rqm.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "linux",
            "enum": [
              "linux"
            ]
          },
```

```
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "Hostname",
          "Platform"
        ]
      },
      "required": [
        "InstanceId",
        "Platform"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-27apldkhqr0ol

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication instance

```
{
```

```
 "$schema": "http://json-schema.org/draft-04/schema#",
 "name": "Create a DMS replication instance",
 "description": "Create a Database Migration Service (DMS) replication instance on
an Amazon EC2 instance in an AMS VPC. Use the replication instance to perform your
database migration. The replication instance provides high availability and failover
support using a Multi-AZ deployment when you select the Multi-AZ option.",
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "minItems": 0,
     "maxItems": 40,
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
```

```
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-3n1j5hdrmiiiuqk6v",
      "type": "string",
      "enum": [
        "stm-3n1j5hdrmiiiuqk6v"
      ],
      "default": "stm-3n1j5hdrmiiiuqk6v"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AllocatedStorage": {
          "type": "integer",
          "description": "The amount of storage, in gigabytes, to be initially
allocated for the replication instance.",
          "default": 50,
          "minimum": 5,
```

```
          "maximum": 6144
        },
        "AutoMinorVersionUpgrade": {
          "type": "string",
          "description": "True if the replication instance should have automatic minor
engine upgrade during the maintenance window. False if it should not.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "AvailabilityZone": {
          "type": "string",
          "description": "The availability zone for the replication instance. Only
applicable if MultiAZ = false.",
          "default": ""
        },
        "EngineVersion": {
          "type": "string",
          "description": "The engine version number of the replication instance, in the
form 2.4.3.",
          "pattern": "[0-9]{1,2}.[0-9]{1,2}.[0-9]{1,2}|^$",
          "default": ""
        },
        "KmsKeyId": {
          "type": "string",
          "description": "The KMS key identifier that will be used to encrypt the
content on the replication instance.",
          "pattern": "^$|^[\\w]{8}-[\\w]{4}-[\\w]{4}-[\\w]{4}-[\\w]{12}$",
          "default": ""
        },
        "MultiAZ": {
          "type": "string",
          "description": "True if the replication instance is a Multi-AZ deployment.
False if it is not.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
```

```
          "description": "The weekly time range during which system maintenance
 can occur, in Universal Coordinated Time (UTC). Must be in the format ddd:hh24:mi-
ddd:hh24:mi, and must be at least 30 minutes.",
          "pattern": "([a-zA-Z]{3}:[0-2]{1}[0-9]{1}:[0-6]{1}[0-9]{1}-[a-zA-Z]{3}:[0-2]
{1}[0-9]{1}:[0-6]{1}[0-9]{1}|)",
          "default": ""
        },
        "InstanceClass": {
          "type": "string",
          "description": "The Amazon EC2 instance class for the replication instance
 to use to perform your database migration, in the form dms.t2.micro. AWS DMS currently
 supports the T2, C4, and R4 Amazon EC2 instance classes for replication instances.",
          "pattern": "dms.[0-9a-z]{2,4}.[0-9a-z]{2,10}",
          "default": "dms.t2.micro"
        },
        "Identifier": {
          "type": "string",
          "description": "The identifier for the replication instance. Given a unique
 ID if none is provided.",
          "pattern": "([a-z][a-z0-9]*(-[a-z0-9]+)*|)",
          "default": ""
        },
        "ReplicationSubnetGroupIdentifier": {
          "type": "string",
          "description": "The subnet group identifier to associate with the replication
 instance.",
          "pattern": "[0-9a-zA-Z\\-]{1,255}"
        },
        "SecurityGroupIds": {
          "type": "array",
          "description": "The identifiers of the security groups to control traffic to
 and from the replication instance. If your source database is in a VPC, select the VPC
 security group that provides access to the DB instance where the database resides.",
          "items": {
            "type": "string"
          }
        }
      },
      "metadata": {
        "ui:order": [
          "Identifier",
          "InstanceClass",
          "AllocatedStorage",
          "EngineVersion",
```

```
          "AutoMinorVersionUpgrade",
          "ReplicationSubnetGroupIdentifier",
          "SecurityGroupIds",
          "AvailabilityZone",
          "MultiAZ",
          "KmsKeyId",
          "PreferredMaintenanceWindow"
        ]
      },
      "required": [
        "InstanceClass",
        "ReplicationSubnetGroupIdentifier",
        "SecurityGroupIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-27jjy5wnrfef2

Classifications:

- Management | Advanced stack components | RDS database stack | Update maintenance window (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update RDS Maintenance Window",
  "type": "object",
  "description": "Update an existing RDS maintenance window, which is a weekly time
 range (in UTC) during which system maintenance can occur. Changing an RDS maintenance
 window doesn't result in an outage. If moving this window to the current time, there
 must be at least 30 minutes between the current time and the end of the current window
 to ensure pending changes are applied.",
  "properties": {
    "DBIdentifierArn": {
      "description": "The Amazon Resource Name (ARN) that uniquely identifies the RDS
 DB instance or cluster.",
      "type": "string",
      "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{12}:(db|cluster):[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
    },
    "PreferredMaintenanceWindow": {
      "type": "string",
      "description": "The weekly time range during which system maintenance can occur,
 in UTC. Must be in the format ddd:hh24:mi-ddd:hh24:mi (Sun:05:00-Sun:05:30), in
 Universal Coordinated Time (UTC) and must be at least 30 minutes. If you don't specify
 PreferredMaintenanceWindow, then Amazon RDS assigns a 30-minute maintenance window on
 a randomly selected day of the week.",
      "pattern": "[a-zA-Z]{3}:[0-9]{2}:[0-9]{2}-[a-zA-Z]{3}:[0-9]{2}:[0-9]{2}$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
```

```
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DBIdentifierArn",
      "PreferredMaintenanceWindow",
      "Priority"
    ]
  },
  "required": [
    "DBIdentifierArn",
    "PreferredMaintenanceWindow"
  ]
}
```

Schema for Change Type ct-27tuth19k52b4

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Update entity or policy (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update IAM Resource",
  "description": "Update Identity and Access Management (IAM) user, role, or policy.",
  "type": "object",
  "properties": {
    "UseCase": {
      "description": "Provide a detailed use case for the IAM user, role, or policy
 change.",
      "type": "string",
      "minLength": 1,
      "maxLength": 1000
    },
    "IAM User": {
      "description": "Update IAM user.",
      "type": "array",
      "items": {
        "type": "object",
```

```
        "properties": {
          "UserName": {
            "description": "The name of the IAM user to modify. The name can be up
 to 64 characters in length, and is limited to use characters a-z, A-Z, 0-9, and _
+=,.@-.",
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$",
            "minLength": 1,
            "maxLength": 64
          },
          "UserPermissions": {
            "description": "Detailed information about the user permissions, or
 a policy document to be attached to the user (paste the policy document into the
 value field). Details should include the type of access (for example Read, Write or
 Delete).",
            "type": "string",
            "minLength": 1,
            "maxLength": 5000
          },
          "Tags": {
            "description": "Up to 50 tags (key/value pairs) to categorize the IAM
 User.",
            "type": "array",
            "items": {
              "type": "object",
              "properties": {
                "Key": {
                  "type": "string",
                  "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                  "minLength": 1,
                  "maxLength": 127
                },
                "Value": {
                  "type": "string",
                  "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                  "minLength": 1,
                  "maxLength": 255
                }
              },
              "additionalProperties": false,
              "metadata": {
                "ui:order": [
                  "Key",
                  "Value"
```

```
              ]
            },
            "required": [
              "Key",
              "Value"
            ]
          },
          "minItems": 0,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "UserName",
          "UserPermissions",
          "Tags"
        ]
      },
      "required": [
        "UserName",
        "UserPermissions"
      ]
    },
    "minItems": 0,
    "maxItems": 1
  },
  "IAM Role": {
    "description": "Update IAM role.",
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "RoleName": {
          "description": "The name of the IAM role to modify. The name can be up
 to 64 characters in length, and is limited to use characters a-z, A-Z, 0-9, and _
+=,.@-.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$",
          "minLength": 1,
          "maxLength": 64
        },
        "TrustPolicy": {
```

```
          "description": "Detailed information about trust relationship, or an assume
role policy document to be attached to the role (paste the policy document into the
value field).",
          "type": "string",
          "minLength": 1,
          "maxLength": 5000
        },
        "RolePermissions": {
          "description": "Detailed information about role permissions, or a policy
document to be attached to the role (paste the policy document into the value field).
Details should include the type of access (for example Read, Write or Delete).",
          "type": "string",
          "minLength": 1,
          "maxLength": 5000
        },
        "Tags": {
          "description": "Up to 50 tags (key/value pairs) to categorize the IAM
role.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "Key": {
                "type": "string",
                "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                "minLength": 1,
                "maxLength": 127
              },
              "Value": {
                "type": "string",
                "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                "minLength": 1,
                "maxLength": 255
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "Key",
                "Value"
              ]
            },
            "required": [
              "Key",
```

```
                "Value"
              ]
            },
            "minItems": 0,
            "maxItems": 50,
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "RoleName",
            "TrustPolicy",
            "RolePermissions",
            "Tags"
          ]
        },
        "required": [
          "RoleName"
        ]
      },
      "minItems": 0,
      "maxItems": 1
    },
    "IAM Policy": {
      "description": "Update IAM policy.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "PolicyName": {
            "description": "The name of the IAM policy to modify. The name can be up
 to 128 characters in length, and is limited to use characters a-z, A-Z, 0-9, and _
+=,.@-.",
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,128}$",
            "minLength": 1,
            "maxLength": 64
          },
          "PolicyDocument": {
            "description": "Detailed information about policy permissions update, or a
 policy document (paste the policy document into the value field).",
            "type": "string",
            "minLength": 1,
```

```
            "maxLength": 20480
          },
          "RelatedResources": {
            "description": "IAM users or roles to which the policy applies.",
            "type": "array",
            "items": {
              "type": "string",
              "minLength": 1,
              "maxLength": 64
            },
            "minItems": 0,
            "maxItems": 10,
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "PolicyName",
            "PolicyDocument",
            "RelatedResources"
          ]
        },
        "required": [
          "PolicyName"
        ]
      },
      "minItems": 0,
      "maxItems": 10,
      "uniqueItems": true
    },
    "Operation": {
      "description": "Must be Update.",
      "type": "string",
      "default": "Update",
      "enum": [
        "Update"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
```

```
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "UseCase",
      "IAM User",
      "IAM Role",
      "IAM Policy",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "UseCase",
    "Operation"
  ]
}
```

Schema for Change Type ct-281dpwh9tqnan

Classifications:

- Deployment | Managed Firewall | Outbound (Palo Alto) | Create security policy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Security Policy",
  "description": "Create a security policy for AMS managed Palo Alto firewall -
 Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be CreateSecurityPolicy.",
      "type": "string",
      "enum": [
        "CreateSecurityPolicy"
      ],
```

```
      "default": "CreateSecurityPolicy"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityPolicyName": {
          "description": "A meaningful name for the security policy. Must start with
 custom-sec-.",
          "type": "string",
          "pattern": "^custom-sec-[a-zA-Z0-9][a-zA-Z0-9-_]{0,51}$"
        },
        "SourceAddresses": {
          "description": "A list of source addresses. If no value is provided, the
 security policy will match against any source address.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "DestinationAddresses": {
          "description": "A list of destination addresses. Supply values for this
 parameter or for AllowLists, but not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9]+\\.[0-9]+\\.[0-9]+\\.[0-9]+(/[0-9]{1,2})?)|((([a-zA-
Z0-9][a-zA-Z0-9-_]{0,62}[a-zA-Z0-9]{0,1}))\\.){1,127}([a-zA-Z][a-zA-Z0-9\\-]{0,23}[a-
zA-Z]))$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "AllowLists": {
          "description": "A list of allowlists to associate with this security policy.
 Supply values for this parameter or for DestinationAddresses, but not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
          },
          "minItems": 1,
```

```
          "maxItems": 10
        },
        "ServicePorts": {
          "type": "object",
          "description": "A list of Transmission Control Protocol (TCP) and User
Datagram Protocol (UDP) service ports. If no value is provided, the security policy
matches against any service port.",
          "properties": {
            "tcp": {
              "description": "A list of Transmission Control Protocol (TCP) service
ports. If no value is provided for TCP or UDP, the security policy matches against any
service port.",
              "type": "array",
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            },
            "udp": {
              "description": "A list of User Datagram Protocol (UDP) service ports. If
no value is provided for TCP or UDP, the security policy matches against any service
port.",
              "type": "array",
              "items": {
                "type": "integer",
                "minimum": 1,
                "maximum": 65535
              },
              "minItems": 1,
              "maxItems": 50
            }
          },
          "metadata": {
            "ui:order": [
              "tcp",
              "udp"
            ]
          }
        },
        "ActionType": {
```

```
          "description": "The type of action the security policy will perform on
 outbound traffic that matches the policy's rules.",
          "type": "string",
          "enum": [
            "Allow",
            "Deny"
          ],
          "default": "Allow"
        },
        "EnablePolicy": {
          "description": "True to enable the security policy upon creation, false to
 not enable it (the policy must be explicitly enabled instead). Default is true.",
          "type": "boolean",
          "default": true
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "SecurityPolicyName",
          "SourceAddresses",
          "DestinationAddresses",
          "AllowLists",
          "ServicePorts",
          "ActionType",
          "EnablePolicy"
        ]
      },
      "required": [
        "SecurityPolicyName",
        "ActionType",
        "EnablePolicy"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "RequestType",
      "Parameters"
    ]
  },
  "required": [
    "RequestType",
```

```
    "Parameters"
  ]
}
```

Schema for Change Type ct-281et7bs9ep4s

Classifications:

- Deployment | Advanced stack components | OpenSearch | Create domain

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create an Amazon OpenSearch Service Domain",
  "description": "Create an Amazon OpenSearch Service domain. An OpenSearch domain
 encapsulates OpenSearch engine instances that process OpenSearch requests. Amazon
 OpenSearch Service supports OpenSearch and legacy Elasticsearch OSS (up to 7.10, the
 final open source version of the software).",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "The ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
```

```
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-szccoe02000000000.",
      "type": "string",
      "enum": [
        "stm-szccoe02000000000"
      ],
      "default": "stm-szccoe02000000000"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
```

```
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DomainName": {
          "type": "string",
          "description": "A name for the OpenSearch Service domain. Domain names
must start with a lowercase letter and must be between 3 and 28 characters. Valid
characters are a-z (lowercase only), 0-9, and û (hyphen).",
          "pattern": "^[a-z][a-z0-9-]{3,28}$"
        },
        "EngineVersion": {
          "type": "string",
          "description": "The version of the OpenSearch Service to use.",
          "enum": [
            "OpenSearch_2.3",
            "OpenSearch_1.3",
            "OpenSearch_1.2",
            "OpenSearch_1.1",
            "OpenSearch_1.0",
            "Elasticsearch_7.10",
            "Elasticsearch_7.9",
            "Elasticsearch_7.8",
            "Elasticsearch_7.7",
            "Elasticsearch_7.4",
            "Elasticsearch_7.1",
            "Elasticsearch_6.8",
            "Elasticsearch_6.7",
            "Elasticsearch_6.5",
            "Elasticsearch_6.4",
            "Elasticsearch_6.3",
            "Elasticsearch_6.2",
            "Elasticsearch_6.0",
            "Elasticsearch_5.6",
            "Elasticsearch_5.5",
            "Elasticsearch_5.3",
            "Elasticsearch_5.1",
            "Elasticsearch_2.3",
            "Elasticsearch_1.5"
          ],
          "default": "OpenSearch_2.3"
        },
        "DedicatedMasterCount": {
```

```
          "type": "string",
          "description": "The number of instances to use for the master node. To
disable the dedicated master node, use 0.",
          "enum": [
            "0",
            "3",
            "5"
          ],
          "default": "3"
        },
        "DedicatedMasterType": {
          "type": "string",
          "description": "The instance type that hosts the dedicated master node. If
DedicatedMasterCount > 0 this value must be specified. Otherwise the value here is
ignored.",
          "enum": [
            "c4.2xlarge.search",
            "c4.4xlarge.search",
            "c4.8xlarge.search",
            "c4.large.search",
            "c4.xlarge.search",
            "c5.18xlarge.search",
            "c5.2xlarge.search",
            "c5.4xlarge.search",
            "c5.9xlarge.search",
            "c5.large.search",
            "c5.xlarge.search",
            "c6g.12xlarge.search",
            "c6g.2xlarge.search",
            "c6g.4xlarge.search",
            "c6g.8xlarge.search",
            "c6g.large.search",
            "c6g.xlarge.search",
            "i2.2xlarge.search",
            "i2.xlarge.search",
            "i3.16xlarge.search",
            "i3.2xlarge.search",
            "i3.4xlarge.search",
            "i3.8xlarge.search",
            "i3.large.search",
            "i3.xlarge.search",
            "m3.2xlarge.search",
            "m3.large.search",
            "m3.medium.search",
```

```
            "m3.xlarge.search",
            "m4.10xlarge.search",
            "m4.2xlarge.search",
            "m4.4xlarge.search",
            "m4.large.search",
            "m4.xlarge.search",
            "m5.12xlarge.search",
            "m5.2xlarge.search",
            "m5.4xlarge.search",
            "m5.large.search",
            "m5.xlarge.search",
            "r3.2xlarge.search",
            "r3.4xlarge.search",
            "r3.8xlarge.search",
            "r3.large.search",
            "r3.xlarge.search",
            "r4.16xlarge.search",
            "r4.2xlarge.search",
            "r4.4xlarge.search",
            "r4.8xlarge.search",
            "r4.large.search",
            "r4.xlarge.search",
            "r5.12xlarge.search",
            "r5.2xlarge.search",
            "r5.4xlarge.search",
            "r5.large.search",
            "r5.xlarge.search",
            "r6g.12xlarge.search",
            "r6g.2xlarge.search",
            "r6g.4xlarge.search",
            "r6g.8xlarge.search",
            "r6g.large.search",
            "r6g.xlarge.search",
            "r6gd.12xlarge.search",
            "r6gd.16xlarge.search",
            "r6gd.2xlarge.search",
            "r6gd.4xlarge.search",
            "r6gd.8xlarge.search",
            "r6gd.large.search",
            "r6gd.xlarge.search",
            "t2.medium.search",
            "t2.small.search",
            "t3.medium.search",
            "t3.small.search"
```

```
          ],
          "default": "r6g.large.search"
        },
        "InstanceType": {
          "type": "string",
          "description": "The instance type to use for data nodes in the domain. Must
 be a supported instance type for the selected OpenSearch Service domain version.",
          "enum": [
            "c4.2xlarge.search",
            "c4.4xlarge.search",
            "c4.8xlarge.search",
            "c4.large.search",
            "c4.xlarge.search",
            "c5.18xlarge.search",
            "c5.2xlarge.search",
            "c5.4xlarge.search",
            "c5.9xlarge.search",
            "c5.large.search",
            "c5.xlarge.search",
            "c6g.12xlarge.search",
            "c6g.2xlarge.search",
            "c6g.4xlarge.search",
            "c6g.8xlarge.search",
            "c6g.large.search",
            "c6g.xlarge.search",
            "i2.2xlarge.search",
            "i2.xlarge.search",
            "i3.16xlarge.search",
            "i3.2xlarge.search",
            "i3.4xlarge.search",
            "i3.8xlarge.search",
            "i3.large.search",
            "i3.xlarge.search",
            "m3.2xlarge.search",
            "m3.large.search",
            "m3.medium.search",
            "m3.xlarge.search",
            "m4.10xlarge.search",
            "m4.2xlarge.search",
            "m4.4xlarge.search",
            "m4.large.search",
            "m4.xlarge.search",
            "m5.12xlarge.search",
            "m5.2xlarge.search",
```

```
            "m5.4xlarge.search",
            "m5.large.search",
            "m5.xlarge.search",
            "r3.2xlarge.search",
            "r3.4xlarge.search",
            "r3.8xlarge.search",
            "r3.large.search",
            "r3.xlarge.search",
            "r4.16xlarge.search",
            "r4.2xlarge.search",
            "r4.4xlarge.search",
            "r4.8xlarge.search",
            "r4.large.search",
            "r4.xlarge.search",
            "r5.12xlarge.search",
            "r5.2xlarge.search",
            "r5.4xlarge.search",
            "r5.large.search",
            "r5.xlarge.search",
            "r6g.12xlarge.search",
            "r6g.2xlarge.search",
            "r6g.4xlarge.search",
            "r6g.8xlarge.search",
            "r6g.large.search",
            "r6g.xlarge.search",
            "r6gd.12xlarge.search",
            "r6gd.16xlarge.search",
            "r6gd.2xlarge.search",
            "r6gd.4xlarge.search",
            "r6gd.8xlarge.search",
            "r6gd.large.search",
            "r6gd.xlarge.search",
            "t2.medium.search",
            "t2.small.search",
            "t3.medium.search",
            "t3.small.search"
          ],
          "default": "r6g.large.search"
        },
        "InstanceCount": {
          "type": "integer",
          "description": "The number of data nodes (instances) to use in the OpenSearch
Service domain. If ZoneAwarenessEnabled=true then InstanceCount must be an even
number.",
```

```
          "default": 2,
          "minimum": 1,
          "maximum": 80
        },
        "ZoneAwarenessEnabled": {
          "type": "string",
          "description": "True to enable zone awareness for the OpenSearch Service
domain; false to not. Default is false. When you enable zone awareness, the OpenSearch
Service allocates the nodes and replica index shards that belong to a cluster across
two Availability Zones (AZs) in the same Region to prevent data loss and minimize
downtime in the event of node or data center failure.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "CognitoEnabled": {
          "description": "True to enable Amazon Cognito authentication for OpenSearch
Dashboards; false to not. Default is false.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "AdvancedSecurityOptionsEnabled": {
          "description": "True to enable fine-grained access control; false to not.
Default is false. For true, also set NodeToNodeEncryption=true and EncryptionKey.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "InternalUserDatabaseEnabled": {
          "description": "True to enable the internal user database; false to not.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ],
```

```
          "default": "false"
        },
        "MasterUserARN": {
          "description": "The Amazon Resource Name (ARN) for the master user. Only
 specify if InternalUserDatabaseEnabled=false in AdvancedSecurityOptions.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:(role|user)/[A-Za-
z0-9_-]+$|^$",
          "default": ""
        },
        "MasterUserName": {
          "description": "The username for the master user. Only specify if
 InternalUserDatabaseEnabled=true in AdvancedSecurityOptions.",
          "type": "string",
          "pattern": "[a-zA-Z][a-zA-Z0-9]{1,16}|^$",
          "default": ""
        },
        "MasterUserPassword": {
          "description": "The password for the master user. The master password
 must be at least 8 characters long and contain at least one uppercase letter,
 one lowercase letter, one number, and one special character. Only specify if
 InternalUserDatabaseEnabled=true in AdvancedSecurityOptions.",
          "type": "string",
          "pattern": "^(?=.{8,}$)(?=.*[a-z])(?=.*[A-Z])(?=.*[0-9])(?=.*\\W).*$|^$",
          "default": "",
          "metadata": {
            "ams:sensitive": true
          }
        },
        "CognitoIAMRole": {
          "description": "The AmazonESCognitoAccess role that allows the OpenSearch
 Service to configure your user pool and identity pool.",
          "type": "string",
          "pattern": "^(arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:role/.+)$|^$",
          "default": ""
        },
        "CognitoUserPoolId": {
          "description": "The Amazon Cognito user pool ID that you want the OpenSearch
 Service to use for OpenSearch Dashboards authentication.",
          "type": "string",
          "pattern": "^[A-Za-z0-9\\-\\=\\@\\,\\.]{1,128}$|^$",
          "default": ""
        },
        "CognitoIdentityPoolId": {
```

```
        "description": "The Amazon Cognito identity pool ID that you want the
OpenSearch Service to use for OpenSearch Dashboards authentication.",
        "type": "string",
        "pattern": "^[A-Za-z0-9\\-\\=\\@\\,\\.]{1,128}$|^$",
        "default": ""
      },
      "NodeToNodeEncryption": {
        "description": "True to enable node-to-node encryption on OpenSearch Service
domains; false to not. Default is true.",
        "type": "string",
        "enum": [
          "true",
          "false"
        ],
        "default": "true"
      },
      "EBSIops": {
        "type": "string",
        "description": "The IOPS for EBS volume. Only applies if EBSVolumeType=io1 or
EBSVolumeType=gp3. The minimum value is 1000. The maximum value is 16000.",
        "pattern": "^$|^[1-9][0-9]{3}$|^1[0-5][0-9]{3}$|^16000$",
        "default": ""
      },
      "EBSThroughput": {
        "type": "string",
        "description": "The throughput for EBS volume. Only applies if
EBSVolumeType=gp3. The minimum value is 125. The maximum value is 1000.",
        "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$",
        "default": ""
      },
      "EBSVolumeSize": {
        "type": "integer",
        "description": "The size, in GB, of the EBS volume for each data node. The
minimum and maximum size of an EBS volume depends on the specified EBSVolumeType and
the instance type to which it is attached. For details, see AWS documentation for EBS
volume size limits.",
        "default": 10,
        "minimum": 10,
        "maximum": 1500
      },
      "EBSVolumeType": {
        "type": "string",
        "description": "The storage type for the data node. Storage type does not
apply for dedicated master nodes.",
```

```
          "enum": [
            "standard",
            "gp3",
            "gp2",
            "io1"
          ],
          "default": "gp3"
        },
        "EncryptionKey": {
          "type": "string",
          "description": "The ID or ARN of the KMS master key to use to encrypt data at
 rest.",
          "pattern": "^$|^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-
f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$",
          "default": ""
        },
        "CustomEndpoint": {
          "description": "The fully qualified URL for your custom endpoint.",
          "type": "string",
          "default": ""
        },
        "CustomEndpointCertificateArn": {
          "description": "The AWS Certificate Manager ARN for your domain's SSL/TLS
 certificate.",
          "type": "string",
          "pattern": "^$|^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:
[0-9]{12}:certificate/[a-z0-9-]+$",
          "default": ""
        },
        "TLSSecurityPolicy": {
          "description": "The minimum transport layer security (TLS) version required
 for traffic to the domain. Valid values are TLS 1.0 or 1.2 (default)",
          "type": "string",
          "enum": [
            "Policy-Min-TLS-1-0-2019-07",
            "Policy-Min-TLS-1-2-2019-07"
          ],
          "default": "Policy-Min-TLS-1-2-2019-07"
        },
        "AutomatedSnapshotStartHour": {
          "type": "string",
          "description": "The hour in UTC during which the service takes an automated
 daily snapshot of the indices in the OpenSearch Service domain. For example, if
```

```
you specify 0, the OpenSearch Service takes an automated snapshot everyday between
midnight and 1 am. You can specify a value between 0 and 23.",
          "pattern": "^$|^([0-9]|1[0-9]|2[0-3])$",
          "default": ""
        },
        "SecurityGroups": {
          "type": "array",
          "description": "Comma-separated list of security group (SG) identifiers.
These control access to the OpenSearch Service domain. Leave blank to add the default
private-only security group from the AMS VPC.",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 0,
          "uniqueItems": true
        },
        "SubnetIds": {
          "type": "array",
          "description": "A list of subnet IDs, in the form of subnet-0123abcd or
subnet-01234567890abcdef, to associate with the VPC endpoints for the domain. If
ZoneAwarenessEnabled=true, provide two subnet IDs, one per zone. Otherwise, provide
only one.",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-f0-9]{8}$|^subnet-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 2,
          "uniqueItems": true
        },
        "AllowExplicitIndex": {
          "type": "string",
          "description": "True to allow explicit references to indices inside the body
of HTTP requests; false to not allow. Setting this property to false prevents users
from bypassing access control for sub-resources. Default=true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "IndicesFieldDataCacheSize": {
          "type": "string",
```

```
          "description": "The percentage of Java heap space that is allocated to field
 data. By default, this setting is unbounded.",
          "pattern": "^$|^([0-9]|[1-9][0-9]|100)$",
          "default": ""
        },
        "MaxClauseCount": {
          "type": "string",
          "description": "The maximum number of allowed boolean clauses in a query. By
 default, this setting is 1024.",
          "pattern": "^$|^[1-9][0-9]*$",
          "default": ""
        },
        "ESApplicationLogs": {
          "description": "The CloudWatch log group to publish the OpenSearch Service
 domain error logs.",
          "type": "string",
          "pattern": "^$|^arn:(aws|aws-cn|aws-us-gov):logs:[a-z]{2}-[a-z]+-\\d{1}:[0-9]
{12}:log-group:[\\.\\-_/#A-Za-z0-9]{1,512}(:\\*)?$",
          "default": ""
        },
        "SearchSlowLogs": {
          "description": "The CloudWatch log group to publish the OpenSearch Service
 domain search slow log.",
          "type": "string",
          "pattern": "^$|^arn:(aws|aws-cn|aws-us-gov):logs:[a-z]{2}-[a-z]+-\\d{1}:[0-9]
{12}:log-group:[\\.\\-_/#A-Za-z0-9]{1,512}(:\\*)?$",
          "default": ""
        },
        "IndexSlowLogs": {
          "description": "The CloudWatch log group to publish the OpenSearch Service
 domain index slow log.",
          "type": "string",
          "pattern": "^$|^arn:(aws|aws-cn|aws-us-gov):logs:[a-z]{2}-[a-z]+-\\d{1}:[0-9]
{12}:log-group:[\\.\\-_/#A-Za-z0-9]{1,512}(:\\*)?$",
          "default": ""
        },
        "AuditLogs": {
          "description": "The CloudWatch log group to publish the OpenSearch Service
 domain audit logs.",
          "type": "string",
          "pattern": "^$|^arn:(aws|aws-cn|aws-us-gov):logs:[a-z]{2}-[a-z]+-\\d{1}:[0-9]
{12}:log-group:[\\.\\-_/#A-Za-z0-9]{1,512}(:\\*)?$",
          "default": ""
        }
```

```
    },
    "metadata": {
      "ui:order": [
        "DomainName",
        "EngineVersion",
        "DedicatedMasterCount",
        "DedicatedMasterType",
        "InstanceCount",
        "InstanceType",
        "EBSIops",
        "EBSThroughput",
        "EBSVolumeSize",
        "EBSVolumeType",
        "CognitoEnabled",
        "CognitoIAMRole",
        "CognitoUserPoolId",
        "CognitoIdentityPoolId",
        "CustomEndpoint",
        "CustomEndpointCertificateArn",
        "TLSSecurityPolicy",
        "ESApplicationLogs",
        "SearchSlowLogs",
        "IndexSlowLogs",
        "AuditLogs",
        "NodeToNodeEncryption",
        "SecurityGroups",
        "SubnetIds",
        "AdvancedSecurityOptionsEnabled",
        "InternalUserDatabaseEnabled",
        "MasterUserARN",
        "MasterUserName",
        "MasterUserPassword",
        "ZoneAwarenessEnabled",
        "EncryptionKey",
        "AutomatedSnapshotStartHour",
        "AllowExplicitIndex",
        "IndicesFieldDataCacheSize",
        "MaxClauseCount"
      ]
    },
    "required": [
      "DomainName",
      "EngineVersion",
      "DedicatedMasterCount",
```

```
        "DedicatedMasterType",
        "InstanceType",
        "InstanceCount",
        "EBSVolumeSize",
        "EBSVolumeType",
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2aaaqid7asjy6

Classifications:

- Management | Advanced stack components | EC2 instance stack | Update DeleteOnTermination (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
```

```
 "name": "Update DeleteOnTermination",
 "description": "Update the EBS volume DeleteOnTermination property of the specified
EC2 instance devices.",
 "type": "object",
 "properties": {
   "InstanceId": {
     "description": "The ID of the EC2 instance, in the form i-1234567890abcdef0.",
     "type": "string",
     "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
   },
   "DeviceNames": {
     "description": "The device name or names, where the volume is attached; for
example, /dev/sdf or xvdg.",
     "type": "array",
     "items": {
       "type": "string",
       "pattern": "^(/dev/sd[a-z][1-15]{0,1})|xvd[a-z]$|/dev/xvd[a-z]$|^$"
     },
     "minItems": 1,
     "maxItems": 17,
     "uniqueItems": true
   },
   "DeleteOnTermination": {
     "description": "True to delete the volume when the instance is terminated, False
to not delete it when the instance is terminated. Default is False.",
     "type": "string",
     "default": "False",
     "enum": [
       "True",
       "False"
     ]
   },
   "Priority": {
     "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
     "type": "string",
     "enum": [
       "Low",
       "Medium",
       "High"
     ]
   }
 },
 "additionalProperties": false,
```

```
  "metadata": {
    "ui:order": [
      "InstanceId",
      "DeviceNames",
      "DeleteOnTermination",
      "Priority"
    ]
  },
  "required": [
    "InstanceId",
    "DeviceNames",
    "DeleteOnTermination"
  ]
}
```

Schema for Change Type ct-2b9q8339bj2sa

Classifications:

- Management | Managed Firewall | Outbound (Palo Alto) | Add URLs

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Allow List URLs",
  "description": "Add allow list URLs for AMS managed Palo Alto firewall - Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be AddURLs.",
      "type": "string",
      "enum": [
        "AddURLs"
      ],
      "default": "AddURLs"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "URLs": {
          "description": "URLs to add to the allow list. URLs must end with a forward
 slash i.e '*.amazon.com/'.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "pattern": "^((\\*|([a-zA-Z0-9][a-zA-Z0-9-_]{0,62}[a-zA-Z0-9]{0,1}))\\.)
{1,127}([a-zA-Z][a-zA-Z0-9\\-]{0,23}[a-zA-Z]\\/)$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "AllowListName": {
          "description": "The name of the allow list.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "URLs",
          "AllowListName"
        ]
      },
      "required": [
        "URLs",
        "AllowListName"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Parameters",
      "RequestType"
    ]
  },
  "required": [
    "Parameters",
    "RequestType"
  ]
}
```

Schema for Change Type ct-2bxelbn765ive

Classifications:

- Management | AMS Resource Scheduler | Schedule | Add


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Resource Scheduler Schedule",
  "description": "Add a new schedule to be used in AMS Resource Scheduler. Schedules
 employ defined periods to determine when the specified resource should run.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddOrUpdateSchedule.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddOrUpdateSchedule"
      ],
      "default": "AWSManagedServices-AddOrUpdateSchedule"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
 solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Action": {
          "description": "Specify the value: add. This explicitly requests that the
 Resource Scheduler schedule be added. The option cannot be left blank; it must be
 add.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "add"
            ],
            "default": "add"
          },
```

```
          "maxItems": 1,
          "minItems": 1
        },
        "Name": {
          "description": "A meaningful name for the schedule. The name must be unique
for this account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,64}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Description": {
          "description": "A meaningful description for the schedule.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,1000}$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Hibernate": {
          "description": "True to hibernate (suspend-to-disk) EC2 instances that are
enabled for hibernation and meet hibernation requirements, false to not. Check the
EC2 console to find out if your instances are enabled for hibernation. Default is
false.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Enforced": {
          "description": "True to enforce the schedule, false to not. When this field
is set to true, the Resource Scheduler will stop a running resource if it is manually
```

```
started outside of the running period, and it will start a resource if it is stopped
manually during the running period. Default is false.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "OverrideStatus": {
          "description": "Override the current schedule action. If set to running,
the instance will be started but not stopped until it is manually stopped. Similarly
when set to stopped, the instance will be stopped but not started automatically
until manually started. There is no default. If left unspecified this setting is not
used.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "running",
              "stopped"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Periods": {
          "description": "A comma-separated list of one or more period names in this
schedule. The name, or names, must match the existing defined periods.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,2000}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "RetainRunning": {
```

```
        "description": "True to prevent the Resource Scheduler from stopping a
resource at the end of a period if the instance was manually started before the
beginning of the period. False to not. Default is false.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        },
        "maxItems": 1,
        "minItems": 1
      },
      "StopNewInstances": {
        "description": "True to stop a resource the first time it is tagged if it
is running outside of the running period. False to not stop the resource. Default is
true.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        },
        "maxItems": 1,
        "minItems": 1
      },
      "SSMMaintenanceWindow": {
        "description": "Comma-separated name or names of one, or more, existing AWS
Systems Manager maintenance windows, to use as the period. First, ensure that the
UseMaintenanceWindow parameter is set to true. Create a maintenance window with the
Deployment | Patching | SSM patch window | Create change type (ct-0el2j07llrxs7).",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "(?!^[-_, ]$)^[A-Za-z0-9-_, ]{1,4096}$|^$"
        },
        "maxItems": 1,
        "minItems": 1
      },
      "TimeZone": {
```

```
        "description": "The name of the time zone, in the form US/Pacific, the
schedule uses. If no time zone is specified then the time zone DefaultTimezone set
when the Resource Scheduler was deployed is used.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "Africa/Abidjan",
            "Africa/Accra",
            "Africa/Addis_Ababa",
            "Africa/Algiers",
            "Africa/Asmara",
            "Africa/Bamako",
            "Africa/Bangui",
            "Africa/Banjul",
            "Africa/Bissau",
            "Africa/Blantyre",
            "Africa/Brazzaville",
            "Africa/Bujumbura",
            "Africa/Cairo",
            "Africa/Casablanca",
            "Africa/Ceuta",
            "Africa/Conakry",
            "Africa/Dakar",
            "Africa/Dar_es_Salaam",
            "Africa/Djibouti",
            "Africa/Douala",
            "Africa/El_Aaiun",
            "Africa/Freetown",
            "Africa/Gaborone",
            "Africa/Harare",
            "Africa/Johannesburg",
            "Africa/Juba",
            "Africa/Kampala",
            "Africa/Khartoum",
            "Africa/Kigali",
            "Africa/Kinshasa",
            "Africa/Lagos",
            "Africa/Libreville",
            "Africa/Lome",
            "Africa/Luanda",
            "Africa/Lubumbashi",
            "Africa/Lusaka",
            "Africa/Malabo",
```

```
                "Africa/Maputo",
                "Africa/Maseru",
                "Africa/Mbabane",
                "Africa/Mogadishu",
                "Africa/Monrovia",
                "Africa/Nairobi",
                "Africa/Ndjamena",
                "Africa/Niamey",
                "Africa/Nouakchott",
                "Africa/Ouagadougou",
                "Africa/Porto-Novo",
                "Africa/Sao_Tome",
                "Africa/Tripoli",
                "Africa/Tunis",
                "Africa/Windhoek",
                "America/Adak",
                "America/Anchorage",
                "America/Anguilla",
                "America/Antigua",
                "America/Araguaina",
                "America/Argentina/Buenos_Aires",
                "America/Argentina/Catamarca",
                "America/Argentina/Cordoba",
                "America/Argentina/Jujuy",
                "America/Argentina/La_Rioja",
                "America/Argentina/Mendoza",
                "America/Argentina/Rio_Gallegos",
                "America/Argentina/Salta",
                "America/Argentina/San_Juan",
                "America/Argentina/San_Luis",
                "America/Argentina/Tucuman",
                "America/Argentina/Ushuaia",
                "America/Aruba",
                "America/Asuncion",
                "America/Atikokan",
                "America/Bahia",
                "America/Bahia_Banderas",
                "America/Barbados",
                "America/Belem",
                "America/Belize",
                "America/Blanc-Sablon",
                "America/Boa_Vista",
                "America/Bogota",
                "America/Boise",
```

```
                "America/Cambridge_Bay",
                "America/Campo_Grande",
                "America/Cancun",
                "America/Caracas",
                "America/Cayenne",
                "America/Cayman",
                "America/Chicago",
                "America/Chihuahua",
                "America/Costa_Rica",
                "America/Creston",
                "America/Cuiaba",
                "America/Curacao",
                "America/Danmarkshavn",
                "America/Dawson",
                "America/Dawson_Creek",
                "America/Denver",
                "America/Detroit",
                "America/Dominica",
                "America/Edmonton",
                "America/Eirunepe",
                "America/El_Salvador",
                "America/Fortaleza",
                "America/Glace_Bay",
                "America/Godthab",
                "America/Goose_Bay",
                "America/Grand_Turk",
                "America/Grenada",
                "America/Guadeloupe",
                "America/Guatemala",
                "America/Guayaquil",
                "America/Guyana",
                "America/Halifax",
                "America/Havana",
                "America/Hermosillo",
                "America/Indiana/Indianapolis",
                "America/Indiana/Knox",
                "America/Indiana/Marengo",
                "America/Indiana/Petersburg",
                "America/Indiana/Tell_City",
                "America/Indiana/Vevay",
                "America/Indiana/Vincennes",
                "America/Indiana/Winamac",
                "America/Inuvik",
                "America/Iqaluit",
```

```
                "America/Jamaica",
                "America/Juneau",
                "America/Kentucky/Louisville",
                "America/Kentucky/Monticello",
                "America/Kralendijk",
                "America/La_Paz",
                "America/Lima",
                "America/Los_Angeles",
                "America/Lower_Princes",
                "America/Maceio",
                "America/Managua",
                "America/Manaus",
                "America/Marigot",
                "America/Martinique",
                "America/Matamoros",
                "America/Mazatlan",
                "America/Menominee",
                "America/Merida",
                "America/Metlakatla",
                "America/Mexico_City",
                "America/Miquelon",
                "America/Moncton",
                "America/Monterrey",
                "America/Montevideo",
                "America/Montreal",
                "America/Montserrat",
                "America/Nassau",
                "America/New_York",
                "America/Nipigon",
                "America/Nome",
                "America/Noronha",
                "America/North_Dakota/Beulah",
                "America/North_Dakota/Center",
                "America/North_Dakota/New_Salem",
                "America/Ojinaga",
                "America/Panama",
                "America/Pangnirtung",
                "America/Paramaribo",
                "America/Phoenix",
                "America/Port-au-Prince",
                "America/Port_of_Spain",
                "America/Porto_Velho",
                "America/Puerto_Rico",
                "America/Rainy_River",
```

```
                "America/Rankin_Inlet",
                "America/Recife",
                "America/Regina",
                "America/Resolute",
                "America/Rio_Branco",
                "America/Santa_Isabel",
                "America/Santarem",
                "America/Santiago",
                "America/Santo_Domingo",
                "America/Sao_Paulo",
                "America/Scoresbysund",
                "America/Sitka",
                "America/St_Barthelemy",
                "America/St_Johns",
                "America/St_Kitts",
                "America/St_Lucia",
                "America/St_Thomas",
                "America/St_Vincent",
                "America/Swift_Current",
                "America/Tegucigalpa",
                "America/Thule",
                "America/Thunder_Bay",
                "America/Tijuana",
                "America/Toronto",
                "America/Tortola",
                "America/Vancouver",
                "America/Whitehorse",
                "America/Winnipeg",
                "America/Yakutat",
                "America/Yellowknife",
                "Antarctica/Casey",
                "Antarctica/Davis",
                "Antarctica/DumontDUrville",
                "Antarctica/Macquarie",
                "Antarctica/Mawson",
                "Antarctica/McMurdo",
                "Antarctica/Palmer",
                "Antarctica/Rothera",
                "Antarctica/Syowa",
                "Antarctica/Vostok",
                "Arctic/Longyearbyen",
                "Asia/Aden",
                "Asia/Almaty",
                "Asia/Amman",
```

```
                "Asia/Anadyr",
                "Asia/Aqtau",
                "Asia/Aqtobe",
                "Asia/Ashgabat",
                "Asia/Baghdad",
                "Asia/Bahrain",
                "Asia/Baku",
                "Asia/Bangkok",
                "Asia/Beirut",
                "Asia/Bishkek",
                "Asia/Brunei",
                "Asia/Choibalsan",
                "Asia/Chongqing",
                "Asia/Colombo",
                "Asia/Damascus",
                "Asia/Dhaka",
                "Asia/Dili",
                "Asia/Dubai",
                "Asia/Dushanbe",
                "Asia/Gaza",
                "Asia/Harbin",
                "Asia/Hebron",
                "Asia/Ho_Chi_Minh",
                "Asia/Hong_Kong",
                "Asia/Hovd",
                "Asia/Irkutsk",
                "Asia/Jakarta",
                "Asia/Jayapura",
                "Asia/Jerusalem",
                "Asia/Kabul",
                "Asia/Kamchatka",
                "Asia/Karachi",
                "Asia/Kashgar",
                "Asia/Kathmandu",
                "Asia/Khandyga",
                "Asia/Kolkata",
                "Asia/Krasnoyarsk",
                "Asia/Kuala_Lumpur",
                "Asia/Kuching",
                "Asia/Kuwait",
                "Asia/Macau",
                "Asia/Magadan",
                "Asia/Makassar",
                "Asia/Manila",
```

```
                "Asia/Muscat",
                "Asia/Nicosia",
                "Asia/Novokuznetsk",
                "Asia/Novosibirsk",
                "Asia/Omsk",
                "Asia/Oral",
                "Asia/Phnom_Penh",
                "Asia/Pontianak",
                "Asia/Pyongyang",
                "Asia/Qatar",
                "Asia/Qyzylorda",
                "Asia/Rangoon",
                "Asia/Riyadh",
                "Asia/Sakhalin",
                "Asia/Samarkand",
                "Asia/Seoul",
                "Asia/Shanghai",
                "Asia/Singapore",
                "Asia/Taipei",
                "Asia/Tashkent",
                "Asia/Tbilisi",
                "Asia/Tehran",
                "Asia/Thimphu",
                "Asia/Tokyo",
                "Asia/Ulaanbaatar",
                "Asia/Urumqi",
                "Asia/Ust-Nera",
                "Asia/Vientiane",
                "Asia/Vladivostok",
                "Asia/Yakutsk",
                "Asia/Yekaterinburg",
                "Asia/Yerevan",
                "Atlantic/Azores",
                "Atlantic/Bermuda",
                "Atlantic/Canary",
                "Atlantic/Cape_Verde",
                "Atlantic/Faroe",
                "Atlantic/Madeira",
                "Atlantic/Reykjavik",
                "Atlantic/South_Georgia",
                "Atlantic/St_Helena",
                "Atlantic/Stanley",
                "Australia/Adelaide",
                "Australia/Brisbane",
```

```
                "Australia/Broken_Hill",
                "Australia/Currie",
                "Australia/Darwin",
                "Australia/Eucla",
                "Australia/Hobart",
                "Australia/Lindeman",
                "Australia/Lord_Howe",
                "Australia/Melbourne",
                "Australia/Perth",
                "Australia/Sydney",
                "Canada/Atlantic",
                "Canada/Central",
                "Canada/Eastern",
                "Canada/Mountain",
                "Canada/Newfoundland",
                "Canada/Pacific",
                "Europe/Amsterdam",
                "Europe/Andorra",
                "Europe/Athens",
                "Europe/Belgrade",
                "Europe/Berlin",
                "Europe/Bratislava",
                "Europe/Brussels",
                "Europe/Bucharest",
                "Europe/Budapest",
                "Europe/Busingen",
                "Europe/Chisinau",
                "Europe/Copenhagen",
                "Europe/Dublin",
                "Europe/Gibraltar",
                "Europe/Guernsey",
                "Europe/Helsinki",
                "Europe/Isle_of_Man",
                "Europe/Istanbul",
                "Europe/Jersey",
                "Europe/Kaliningrad",
                "Europe/Kiev",
                "Europe/Lisbon",
                "Europe/Ljubljana",
                "Europe/London",
                "Europe/Luxembourg",
                "Europe/Madrid",
                "Europe/Malta",
                "Europe/Mariehamn",
```

```
                "Europe/Minsk",
                "Europe/Monaco",
                "Europe/Moscow",
                "Europe/Oslo",
                "Europe/Paris",
                "Europe/Podgorica",
                "Europe/Prague",
                "Europe/Riga",
                "Europe/Rome",
                "Europe/Samara",
                "Europe/San_Marino",
                "Europe/Sarajevo",
                "Europe/Simferopol",
                "Europe/Skopje",
                "Europe/Sofia",
                "Europe/Stockholm",
                "Europe/Tallinn",
                "Europe/Tirane",
                "Europe/Uzhgorod",
                "Europe/Vaduz",
                "Europe/Vatican",
                "Europe/Vienna",
                "Europe/Vilnius",
                "Europe/Volgograd",
                "Europe/Warsaw",
                "Europe/Zagreb",
                "Europe/Zaporozhye",
                "Europe/Zurich",
                "GMT",
                "Indian/Antananarivo",
                "Indian/Chagos",
                "Indian/Christmas",
                "Indian/Cocos",
                "Indian/Comoro",
                "Indian/Kerguelen",
                "Indian/Mahe",
                "Indian/Maldives",
                "Indian/Mauritius",
                "Indian/Mayotte",
                "Indian/Reunion",
                "Pacific/Apia",
                "Pacific/Auckland",
                "Pacific/Chatham",
                "Pacific/Chuuk",
```

```
            "Pacific/Easter",
            "Pacific/Efate",
            "Pacific/Enderbury",
            "Pacific/Fakaofo",
            "Pacific/Fiji",
            "Pacific/Funafuti",
            "Pacific/Galapagos",
            "Pacific/Gambier",
            "Pacific/Guadalcanal",
            "Pacific/Guam",
            "Pacific/Honolulu",
            "Pacific/Johnston",
            "Pacific/Kiritimati",
            "Pacific/Kosrae",
            "Pacific/Kwajalein",
            "Pacific/Majuro",
            "Pacific/Marquesas",
            "Pacific/Midway",
            "Pacific/Nauru",
            "Pacific/Niue",
            "Pacific/Norfolk",
            "Pacific/Noumea",
            "Pacific/Pago_Pago",
            "Pacific/Palau",
            "Pacific/Pitcairn",
            "Pacific/Pohnpei",
            "Pacific/Port_Moresby",
            "Pacific/Rarotonga",
            "Pacific/Saipan",
            "Pacific/Tahiti",
            "Pacific/Tarawa",
            "Pacific/Tongatapu",
            "Pacific/Wake",
            "Pacific/Wallis",
            "US/Alaska",
            "US/Arizona",
            "US/Central",
            "US/Eastern",
            "US/Hawaii",
            "US/Mountain",
            "US/Pacific",
            "UTC"
          ]
        },
```

```
          "maxItems": 1,
          "minItems": 1
        },
        "UseMaintenanceWindow": {
          "description": "True to add an Amazon RDS maintenance window as a period to
an Amazon RDS instance schedule, or to add an AWS Systems Manager (SSM) maintenance
window as a period to an Amazon EC2 instance schedule. An RDS maintenance window is
automatically created by RDS. An SSM maintenance window you create with the Deployment
| Patching | SSM maintenance window | Create (ct-0el2j07llrxs7) change type. False
to not add either maintenance window, but to use the start and stop settings of the
period.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "UseMetrics": {
          "description": "Enable CloudWatch metrics for this schedule. This field
overrides the default settings defined when the Resource Scheduler was deployed.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Action",
          "Name",
          "Description",
          "Hibernate",
          "Enforced",
```

```
          "OverrideStatus",
          "Periods",
          "RetainRunning",
          "StopNewInstances",
          "SSMMaintenanceWindow",
          "TimeZone",
          "UseMaintenanceWindow",
          "UseMetrics"
        ]
      },
      "required": [
        "Action",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2c7ve50jost1v

Classifications:

- Management | AMS Resource Scheduler | Solution | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update AMS Resource Scheduler",
  "description": "Update the AMS Resource Scheduler solution in the account.",
```

```
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAMSResourceSchedulerStack-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
      ],
      "default": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SchedulingActive": {
          "description": "Yes to enable the Resource Scheduler. No to disable
 it. The default is existing state. You can also use Resource Scheduler enable
 (ct-2wrvu4kca9xky) and disable (ct-14v49adibs4db) change types to manage its state.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "enum": [
              "Yes",
              "No",
              ""
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ScheduledServices": {
          "description": "Comma-separated list of scheduled services. Use a combination
 of AutoScaling, EC2, and RDS.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
```

```
            "pattern": "^$|(^(ec2|rds|autoscaling)(,(ec2|rds|autoscaling)){0,2}$)"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TagName": {
          "description": "The name of the tag key to use to associate the instance
 schedule schemas with service resources. Default is Schedule.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": "^$|^(?!(aws:|ams:))[a-zA-Z0-9+-=._:/@]{1,127}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "UseCMK": {
          "description": "Comma-separated list of Customer Managed Key (CMK) Amazon
 Resource Names (ARNs) in format arn:<partition>:kms:<region>:<account-id>:key/<key-id>
 to grant Resource Scheduler permission to. These are CMK that are used to encrypt EBS
 volumes on EC2 instances.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(|arn:(aws|aws-cn|aws-us-gov):kms:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{0,12}:key/[a-z0-9\\-]+)$"
          },
          "minItems": 1,
          "maxItems": 20
        },
        "UseLicenseManager": {
          "description": "Comma-separated list of AWS License Manager license ARNs to
 grant Resource Scheduler permission to. These are software or vendor licenses that EC2
 instances are configured with.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(|arn:(aws|aws-cn|aws-us-gov):license-manager:([a-z]{2}((-
gov))?-[a-z]+-\\d{1}):[0-9]{0,12}:license-configuration(/|:)lic-.*)$"
          },
          "minItems": 1,
          "maxItems": 20
        },
```

```
        "DefaultTimezone": {
          "description": "The name of the Time Zone, in the form US/Pacific, to be used
 as default timezone applied. Default is UTC.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "enum": [
              "Africa/Abidjan",
              "Africa/Accra",
              "Africa/Addis_Ababa",
              "Africa/Algiers",
              "Africa/Asmara",
              "Africa/Bamako",
              "Africa/Bangui",
              "Africa/Banjul",
              "Africa/Bissau",
              "Africa/Blantyre",
              "Africa/Brazzaville",
              "Africa/Bujumbura",
              "Africa/Cairo",
              "Africa/Casablanca",
              "Africa/Ceuta",
              "Africa/Conakry",
              "Africa/Dakar",
              "Africa/Dar_es_Salaam",
              "Africa/Djibouti",
              "Africa/Douala",
              "Africa/El_Aaiun",
              "Africa/Freetown",
              "Africa/Gaborone",
              "Africa/Harare",
              "Africa/Johannesburg",
              "Africa/Juba",
              "Africa/Kampala",
              "Africa/Khartoum",
              "Africa/Kigali",
              "Africa/Kinshasa",
              "Africa/Lagos",
              "Africa/Libreville",
              "Africa/Lome",
              "Africa/Luanda",
              "Africa/Lubumbashi",
              "Africa/Lusaka",
```

```
                "Africa/Malabo",
                "Africa/Maputo",
                "Africa/Maseru",
                "Africa/Mbabane",
                "Africa/Mogadishu",
                "Africa/Monrovia",
                "Africa/Nairobi",
                "Africa/Ndjamena",
                "Africa/Niamey",
                "Africa/Nouakchott",
                "Africa/Ouagadougou",
                "Africa/Porto-Novo",
                "Africa/Sao_Tome",
                "Africa/Tripoli",
                "Africa/Tunis",
                "Africa/Windhoek",
                "America/Adak",
                "America/Anchorage",
                "America/Anguilla",
                "America/Antigua",
                "America/Araguaina",
                "America/Argentina/Buenos_Aires",
                "America/Argentina/Catamarca",
                "America/Argentina/Cordoba",
                "America/Argentina/Jujuy",
                "America/Argentina/La_Rioja",
                "America/Argentina/Mendoza",
                "America/Argentina/Rio_Gallegos",
                "America/Argentina/Salta",
                "America/Argentina/San_Juan",
                "America/Argentina/San_Luis",
                "America/Argentina/Tucuman",
                "America/Argentina/Ushuaia",
                "America/Aruba",
                "America/Asuncion",
                "America/Atikokan",
                "America/Bahia",
                "America/Bahia_Banderas",
                "America/Barbados",
                "America/Belem",
                "America/Belize",
                "America/Blanc-Sablon",
                "America/Boa_Vista",
                "America/Bogota",
```

```
                "America/Boise",
                "America/Cambridge_Bay",
                "America/Campo_Grande",
                "America/Cancun",
                "America/Caracas",
                "America/Cayenne",
                "America/Cayman",
                "America/Chicago",
                "America/Chihuahua",
                "America/Costa_Rica",
                "America/Creston",
                "America/Cuiaba",
                "America/Curacao",
                "America/Danmarkshavn",
                "America/Dawson",
                "America/Dawson_Creek",
                "America/Denver",
                "America/Detroit",
                "America/Dominica",
                "America/Edmonton",
                "America/Eirunepe",
                "America/El_Salvador",
                "America/Fortaleza",
                "America/Glace_Bay",
                "America/Godthab",
                "America/Goose_Bay",
                "America/Grand_Turk",
                "America/Grenada",
                "America/Guadeloupe",
                "America/Guatemala",
                "America/Guayaquil",
                "America/Guyana",
                "America/Halifax",
                "America/Havana",
                "America/Hermosillo",
                "America/Indiana/Indianapolis",
                "America/Indiana/Knox",
                "America/Indiana/Marengo",
                "America/Indiana/Petersburg",
                "America/Indiana/Tell_City",
                "America/Indiana/Vevay",
                "America/Indiana/Vincennes",
                "America/Indiana/Winamac",
                "America/Inuvik",
```

```
                "America/Iqaluit",
                "America/Jamaica",
                "America/Juneau",
                "America/Kentucky/Louisville",
                "America/Kentucky/Monticello",
                "America/Kralendijk",
                "America/La_Paz",
                "America/Lima",
                "America/Los_Angeles",
                "America/Lower_Princes",
                "America/Maceio",
                "America/Managua",
                "America/Manaus",
                "America/Marigot",
                "America/Martinique",
                "America/Matamoros",
                "America/Mazatlan",
                "America/Menominee",
                "America/Merida",
                "America/Metlakatla",
                "America/Mexico_City",
                "America/Miquelon",
                "America/Moncton",
                "America/Monterrey",
                "America/Montevideo",
                "America/Montreal",
                "America/Montserrat",
                "America/Nassau",
                "America/New_York",
                "America/Nipigon",
                "America/Nome",
                "America/Noronha",
                "America/North_Dakota/Beulah",
                "America/North_Dakota/Center",
                "America/North_Dakota/New_Salem",
                "America/Ojinaga",
                "America/Panama",
                "America/Pangnirtung",
                "America/Paramaribo",
                "America/Phoenix",
                "America/Port-au-Prince",
                "America/Port_of_Spain",
                "America/Porto_Velho",
                "America/Puerto_Rico",
```

```
                "America/Rainy_River",
                "America/Rankin_Inlet",
                "America/Recife",
                "America/Regina",
                "America/Resolute",
                "America/Rio_Branco",
                "America/Santa_Isabel",
                "America/Santarem",
                "America/Santiago",
                "America/Santo_Domingo",
                "America/Sao_Paulo",
                "America/Scoresbysund",
                "America/Sitka",
                "America/St_Barthelemy",
                "America/St_Johns",
                "America/St_Kitts",
                "America/St_Lucia",
                "America/St_Thomas",
                "America/St_Vincent",
                "America/Swift_Current",
                "America/Tegucigalpa",
                "America/Thule",
                "America/Thunder_Bay",
                "America/Tijuana",
                "America/Toronto",
                "America/Tortola",
                "America/Vancouver",
                "America/Whitehorse",
                "America/Winnipeg",
                "America/Yakutat",
                "America/Yellowknife",
                "Antarctica/Casey",
                "Antarctica/Davis",
                "Antarctica/DumontDUrville",
                "Antarctica/Macquarie",
                "Antarctica/Mawson",
                "Antarctica/McMurdo",
                "Antarctica/Palmer",
                "Antarctica/Rothera",
                "Antarctica/Syowa",
                "Antarctica/Vostok",
                "Arctic/Longyearbyen",
                "Asia/Aden",
                "Asia/Almaty",
```

```
                "Asia/Amman",
                "Asia/Anadyr",
                "Asia/Aqtau",
                "Asia/Aqtobe",
                "Asia/Ashgabat",
                "Asia/Baghdad",
                "Asia/Bahrain",
                "Asia/Baku",
                "Asia/Bangkok",
                "Asia/Beirut",
                "Asia/Bishkek",
                "Asia/Brunei",
                "Asia/Choibalsan",
                "Asia/Chongqing",
                "Asia/Colombo",
                "Asia/Damascus",
                "Asia/Dhaka",
                "Asia/Dili",
                "Asia/Dubai",
                "Asia/Dushanbe",
                "Asia/Gaza",
                "Asia/Harbin",
                "Asia/Hebron",
                "Asia/Ho_Chi_Minh",
                "Asia/Hong_Kong",
                "Asia/Hovd",
                "Asia/Irkutsk",
                "Asia/Jakarta",
                "Asia/Jayapura",
                "Asia/Jerusalem",
                "Asia/Kabul",
                "Asia/Kamchatka",
                "Asia/Karachi",
                "Asia/Kashgar",
                "Asia/Kathmandu",
                "Asia/Khandyga",
                "Asia/Kolkata",
                "Asia/Krasnoyarsk",
                "Asia/Kuala_Lumpur",
                "Asia/Kuching",
                "Asia/Kuwait",
                "Asia/Macau",
                "Asia/Magadan",
                "Asia/Makassar",
```

```
                "Asia/Manila",
                "Asia/Muscat",
                "Asia/Nicosia",
                "Asia/Novokuznetsk",
                "Asia/Novosibirsk",
                "Asia/Omsk",
                "Asia/Oral",
                "Asia/Phnom_Penh",
                "Asia/Pontianak",
                "Asia/Pyongyang",
                "Asia/Qatar",
                "Asia/Qyzylorda",
                "Asia/Rangoon",
                "Asia/Riyadh",
                "Asia/Sakhalin",
                "Asia/Samarkand",
                "Asia/Seoul",
                "Asia/Shanghai",
                "Asia/Singapore",
                "Asia/Taipei",
                "Asia/Tashkent",
                "Asia/Tbilisi",
                "Asia/Tehran",
                "Asia/Thimphu",
                "Asia/Tokyo",
                "Asia/Ulaanbaatar",
                "Asia/Urumqi",
                "Asia/Ust-Nera",
                "Asia/Vientiane",
                "Asia/Vladivostok",
                "Asia/Yakutsk",
                "Asia/Yekaterinburg",
                "Asia/Yerevan",
                "Atlantic/Azores",
                "Atlantic/Bermuda",
                "Atlantic/Canary",
                "Atlantic/Cape_Verde",
                "Atlantic/Faroe",
                "Atlantic/Madeira",
                "Atlantic/Reykjavik",
                "Atlantic/South_Georgia",
                "Atlantic/St_Helena",
                "Atlantic/Stanley",
                "Australia/Adelaide",
```

```
                "Australia/Brisbane",
                "Australia/Broken_Hill",
                "Australia/Currie",
                "Australia/Darwin",
                "Australia/Eucla",
                "Australia/Hobart",
                "Australia/Lindeman",
                "Australia/Lord_Howe",
                "Australia/Melbourne",
                "Australia/Perth",
                "Australia/Sydney",
                "Canada/Atlantic",
                "Canada/Central",
                "Canada/Eastern",
                "Canada/Mountain",
                "Canada/Newfoundland",
                "Canada/Pacific",
                "Europe/Amsterdam",
                "Europe/Andorra",
                "Europe/Athens",
                "Europe/Belgrade",
                "Europe/Berlin",
                "Europe/Bratislava",
                "Europe/Brussels",
                "Europe/Bucharest",
                "Europe/Budapest",
                "Europe/Busingen",
                "Europe/Chisinau",
                "Europe/Copenhagen",
                "Europe/Dublin",
                "Europe/Gibraltar",
                "Europe/Guernsey",
                "Europe/Helsinki",
                "Europe/Isle_of_Man",
                "Europe/Istanbul",
                "Europe/Jersey",
                "Europe/Kaliningrad",
                "Europe/Kiev",
                "Europe/Lisbon",
                "Europe/Ljubljana",
                "Europe/London",
                "Europe/Luxembourg",
                "Europe/Madrid",
                "Europe/Malta",
```

```
                "Europe/Mariehamn",
                "Europe/Minsk",
                "Europe/Monaco",
                "Europe/Moscow",
                "Europe/Oslo",
                "Europe/Paris",
                "Europe/Podgorica",
                "Europe/Prague",
                "Europe/Riga",
                "Europe/Rome",
                "Europe/Samara",
                "Europe/San_Marino",
                "Europe/Sarajevo",
                "Europe/Simferopol",
                "Europe/Skopje",
                "Europe/Sofia",
                "Europe/Stockholm",
                "Europe/Tallinn",
                "Europe/Tirane",
                "Europe/Uzhgorod",
                "Europe/Vaduz",
                "Europe/Vatican",
                "Europe/Vienna",
                "Europe/Vilnius",
                "Europe/Volgograd",
                "Europe/Warsaw",
                "Europe/Zagreb",
                "Europe/Zaporozhye",
                "Europe/Zurich",
                "GMT",
                "Indian/Antananarivo",
                "Indian/Chagos",
                "Indian/Christmas",
                "Indian/Cocos",
                "Indian/Comoro",
                "Indian/Kerguelen",
                "Indian/Mahe",
                "Indian/Maldives",
                "Indian/Mauritius",
                "Indian/Mayotte",
                "Indian/Reunion",
                "Pacific/Apia",
                "Pacific/Auckland",
                "Pacific/Chatham",
```

```
                "Pacific/Chuuk",
                "Pacific/Easter",
                "Pacific/Efate",
                "Pacific/Enderbury",
                "Pacific/Fakaofo",
                "Pacific/Fiji",
                "Pacific/Funafuti",
                "Pacific/Galapagos",
                "Pacific/Gambier",
                "Pacific/Guadalcanal",
                "Pacific/Guam",
                "Pacific/Honolulu",
                "Pacific/Johnston",
                "Pacific/Kiritimati",
                "Pacific/Kosrae",
                "Pacific/Kwajalein",
                "Pacific/Majuro",
                "Pacific/Marquesas",
                "Pacific/Midway",
                "Pacific/Nauru",
                "Pacific/Niue",
                "Pacific/Norfolk",
                "Pacific/Noumea",
                "Pacific/Pago_Pago",
                "Pacific/Palau",
                "Pacific/Pitcairn",
                "Pacific/Pohnpei",
                "Pacific/Port_Moresby",
                "Pacific/Rarotonga",
                "Pacific/Saipan",
                "Pacific/Tahiti",
                "Pacific/Tarawa",
                "Pacific/Tongatapu",
                "Pacific/Wake",
                "Pacific/Wallis",
                "US/Alaska",
                "US/Arizona",
                "US/Central",
                "US/Eastern",
                "US/Hawaii",
                "US/Mountain",
                "US/Pacific",
                "UTC",
                ""
```

```
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Action": {
          "description": "Must be Update.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Update"
            ],
            "default": "Update"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SchedulingActive",
          "ScheduledServices",
          "TagName",
          "DefaultTimezone",
          "UseCMK",
          "UseLicenseManager",
          "Action"
        ]
      },
      "required": [
        "Action"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
```

```
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2d55p1d7z6w3d

Classifications:

- Management | Advanced stack components | EBS Volume | Detach

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Detach EBS Volume",
  "description": "Detach an EBS volume from an EC2 instance. This change type provides
 an option that attempts to remediate drift in the CloudFormation stack where the
 volume is being detached, but that option, RemediateStackDrift, does not work on
 volumes created using the CloudFormation ingest change type (ct-36cn2avfrrj9v).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DetachEBSVolume.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DetachEBSVolume"
      ],
      "default": "AWSManagedServices-DetachEBSVolume"
    },
    "Region": {
      "description": "The AWS Region where the EBS Volume is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VolumeId": {
          "description": "The ID of the EBS volume, in the form
 vol-1234567890abcdef0.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^vol-([0-9a-f]{8}|[0-9a-f]{17})$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused
by volume modification. False to not attempt drift remediation. Drift remediation can
be performed only on CloudFormation stacks that were created using a CT other than
the Ingestion CT ct-36cn2avfrrj9v and that are in sync with the definitions in the
stack template prior to the volume modification. Set to False to modify a volume in an
ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "VolumeId",
          "RemediateStackDrift"
        ]
      },
      "required": [
        "VolumeId"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
```

```
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2dphvdy1krpj6

Classifications:

- Management | Advanced stack components | RDS database stack | Update (for Aurora)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update RDS Aurora stack",
  "description": "Modify the properties of an existing AWS Relational Database Service
 (RDS) Aurora stack created using CT ID ct-2jvzjwunghrhy, version 1.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the RDS Aurora cluster you are updating, in the
 form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AutoMinorVersionUpgrade": {
          "type": "string",
```

```
          "description": "True if the RDS instance should have automatic minor version
 upgrade, false if it should not.",
          "enum": [
            "true",
            "false"
          ]
        },
        "BackupRetentionPeriod": {
          "type": "integer",
          "description": "The number of days for which automatic database (DB)
 snapshots are retained. Range is 1 - 35.",
          "minimum": 1,
          "maximum": 35
        },
        "EngineVersion": {
          "type": "string",
          "description": "The version number of the database engine to use. Not every
 database version is available for every AWS region.",
          "pattern": "^\\d.\\d.\\d{2}[a-z]$|^5.\\d.mysql_aurora.\\d.\\d{2}.\\d$|^8.\
\d.mysql_aurora.\\d.\\d{2}.\\d$|^(\\d{2}.\\d{0,2})$|^$"
        },
        "InstanceType": {
          "type": "string",
          "description": "The instance type to use, this determines the compute and
 memory capacity for the DB instance. Not every instance type is available for every
 database engine.",
          "enum": [
            "db.serverless",
            "db.t2.small",
            "db.t2.medium",
            "db.t3.micro",
            "db.t3.small",
            "db.t3.medium",
            "db.t3.large",
            "db.t3.xlarge",
            "db.t3.2xlarge",
            "db.t4g.medium",
            "db.t4g.large",
            "db.r3.large",
            "db.r3.xlarge",
            "db.r3.2xlarge",
            "db.r3.4xlarge",
            "db.r3.8xlarge",
            "db.r4.large",
```

```
            "db.r4.xlarge",
            "db.r4.2xlarge",
            "db.r4.4xlarge",
            "db.r4.8xlarge",
            "db.r4.16xlarge",
            "db.r5.large",
            "db.r5.xlarge",
            "db.r5.2xlarge",
            "db.r5.4xlarge",
            "db.r5.8xlarge",
            "db.r5.12xlarge",
            "db.r5.16xlarge",
            "db.r5.24xlarge",
            "db.r6g.large",
            "db.r6g.xlarge",
            "db.r6g.2xlarge",
            "db.r6g.4xlarge",
            "db.r6g.8xlarge",
            "db.r6g.12xlarge",
            "db.r6g.16xlarge",
            "db.x2g.large",
            "db.x2g.xlarge",
            "db.x2g.2xlarge",
            "db.x2g.4xlarge",
            "db.x2g.8xlarge",
            "db.x2g.12xlarge",
            "db.x2g.16xlarge"
          ]
        },
        "MasterUserPassword": {
          "type": "string",
          "description": "The password that you use with the configured MasterUsername
to log in to your DB instance. Must contain from 8 to 41 printable ASCII characters
(excluding backslash, double quotes, and at sign).",
          "pattern": "^$|(?!@/\")[a-zA-Z0-9]{8,41}$",
          "maxLength": 41,
          "minLength": 8,
          "metadata": {
            "ams:sensitive": true
          }
        },
        "MultiAZ": {
          "type": "string",
```

```
          "description": "True to have a secondary replica of your DB instance created
 in another Availability Zone for failover support, false to not have a standby.",
          "enum": [
            "true",
            "false"
          ]
        },
        "PerformanceInsights": {
          "type": "string",
          "description": "True to enable Performance Insights for the DB instance,
 false to not. Performance Insights is only available on engine type aurora and aurora-
postgresql.",
          "enum": [
            "true",
            "false"
          ]
        },
        "PerformanceInsightsKMSKey": {
          "type": "string",
          "description": "ARN of the KMS master key to use to encrypt Performance
 Insights data.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
        },
        "PerformanceInsightsRetentionPeriod": {
          "type": "string",
          "description": "The amount of time, in days, to retain Performance Insights
 data. Valid values are 7 or 731 (2 years).",
          "enum": [
            "7",
            "731"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number on which the database accepts connections.
 Valid range is: 1150-65535.",
          "pattern": "^(0|11[5-8][0-9]|119[0-9]|1[2-9][0-9]{2}|[2-9][0-9]{3}|[1-5][0-9]
{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])$"
        },
        "PreferredBackupWindow": {
          "type": "string",
          "description": "The daily time range during which automated backups are
 created. Must be in the format hh:mm-hh:mm (24-hour format), in Universal Coordinated
```

```
Time (UTC). Must not conflict with the PreferredMaintenanceWindow setting, and must be
at least 30 minutes.",
          "pattern": "^[0-9]{2}:[0-9]{2}-[0-9]{2}:[0-9]{2}$"
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
          "description": "The weekly time range during which system maintenance
can occur, in UTC. Must be in the format ddd:hh:mm-ddd:hh:mm (24-hour format), in
Universal Coordinated Time (UTC) and must be at least 30 minutes. If you don't specify
PreferredMaintenanceWindow, then Amazon RDS assigns a 30-minute maintenance window on
a randomly selected day of the week.",
          "pattern": "^$|[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$"
        },
        "ServerlessScalingMaxCapacity": {
          "description": "The maximum number of Aurora capacity units (ACUs) for a DB
instance in an Aurora Serverless cluster. The largest value that you can use is 128.0.
Only applies to db.serverless InstanceType.",
          "type": "number",
          "minimum": 1,
          "maximum": 128
        },
        "ServerlessScalingMinCapacity": {
          "description": "The minimum number of Aurora capacity units (ACUs) for a DB
instance in an Aurora Serverless cluster. The smallest  value that you can use is 0.5.
Only applies to db.serverless InstanceType.",
          "type": "number",
          "minimum": 0.5,
          "maximum": 128
        }
      },
      "metadata": {
        "ui:order": [
          "EngineVersion",
          "InstanceType",
          "MultiAZ",
          "MasterUserPassword",
          "Port",
          "AutoMinorVersionUpgrade",
          "PerformanceInsights",
          "PerformanceInsightsKMSKey",
          "PerformanceInsightsRetentionPeriod",
          "BackupRetentionPeriod",
          "PreferredBackupWindow",
          "PreferredMaintenanceWindow",
```

```
          "ServerlessScalingMaxCapacity",
          "ServerlessScalingMinCapacity"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2edc3sd1sqmrb

Classifications:

- Deployment | Applications | CodeDeploy application | Deploy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Deploy CodeDeploy Application",
  "description": "Deploy a revision of an existing AWS CodeDeploy application, which
 are source files CodeDeploy will deploy to your instances or scripts CodeDeploy will
 run on your instances.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The reason for the request.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
```

```
   "VpcId": {
     "description": "Identifier of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "TimeoutInMinutes": {
     "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
     "type": "number",
     "minimum": 0,
     "maximum": 360
   },
   "Parameters": {
     "description": "Specifications for the deployment.",
     "type": "object",
     "properties": {
       "CodeDeployApplicationName": {
         "description": "The name of the AWS CodeDeploy application.",
         "type": "string",
         "minLength": 1,
         "maxLength": 100,
         "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
       },
       "CodeDeployDeploymentConfigName": {
         "description": "The configuration for deployment operations: as many
instances as possible at once, half of the instances at a time, or only one instance
at a time.",
         "type": "string",
         "enum": [
           "CodeDeployDefault.AllAtOnce",
           "CodeDeployDefault.HalfAtATime",
           "CodeDeployDefault.OneAtATime"
         ],
         "default": "CodeDeployDefault.OneAtATime"
       },
```

```
        "CodeDeployDeploymentGroupName": {
          "description": "The name of the deployment group.",
          "type": "string",
          "minLength": 1,
          "maxLength": 100,
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        },
        "CodeDeployIgnoreApplicationStopFailures": {
          "description": "True to ignore the failure of an ApplicationStop lifecycle
event and continue to the BeforeInstall event; false to stop the deployment if the
ApplicationStop event fails. Default is false.",
          "type": "boolean",
          "default": false
        },
        "CodeDeployRevision": {
          "description": "The type and location of the revision to deploy.",
          "type": "object",
          "properties": {
            "RevisionType": {
              "type": "string",
              "enum": [
                "S3"
              ]
            },
            "S3Location": {
              "type": "object",
              "properties": {
                "S3Bucket": {
                  "description": "The name of the Amazon S3 bucket where the
application revision is stored.",
                  "type": "string"
                },
                "S3BundleType": {
                  "description": "The file type of the application revision.",
                  "type": "string",
                  "enum": [
                    "tar",
                    "tgz",
                    "zip"
                  ]
                },
                "S3ETag": {
                  "description": "The ETag of the Amazon S3 object that represents the
bundled artifacts for the application revision.",
```

```
                  "type": "string"
                },
                "S3Key": {
                  "description": "The name of the Amazon S3 object that represents
 the bundled artifacts for the application revision (e.g. my_app.zip or path/to/
my_app.zip).",
                  "type": "string"
                },
                "S3Version": {
                  "description": "A specific version of the Amazon S3 object that
 represents the bundled artifacts for the application revision.",
                  "type": "string"
                }
              },
              "additionalProperties": false,
              "metadata": {
                "ui:order": [
                  "S3Bucket",
                  "S3BundleType",
                  "S3Key",
                  "S3ETag",
                  "S3Version"
                ]
              },
              "required": [
                "S3Bucket",
                "S3BundleType",
                "S3Key"
              ]
            }
          },
          "additionalProperties": false,
          "metadata": {
            "ui:order": [
              "RevisionType",
              "S3Location"
            ]
          },
          "required": [
            "RevisionType",
            "S3Location"
          ]
        }
      },
```

```
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "CodeDeployApplicationName",
          "CodeDeployDeploymentConfigName",
          "CodeDeployDeploymentGroupName",
          "CodeDeployIgnoreApplicationStopFailures",
          "CodeDeployRevision"
        ]
      },
      "required": [
        "CodeDeployApplicationName",
        "CodeDeployDeploymentGroupName",
        "CodeDeployRevision"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2eof6j3mlcwhf

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Service-Linked role

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Service-Linked Role",
  "description": "Create an IAM service-linked role linked to an AWS service that you
 specify.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateServiceLinkedRole-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateServiceLinkedRole-Admin"
      ],
      "default": "AWSManagedServices-CreateServiceLinkedRole-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AWSServiceName": {
          "description": "The service principal, in the form <service-principal-
name>.amazonaws.com. This value becomes the Principal element in the policy for
 the role. To verify that an AWS service supports IAM service-linked roles, see:
 AWS services that work with IAM. For a list of service principal names, see GitHub
 Gist: List of AWS Service Principals. Example: EC2 Auto Scaling service principal is
 autoscaling.amazonaws.com.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z-.\\d]{2,}.amazonaws.com$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CustomSuffix": {
          "description": "A string that you provide, which is combined with the
 service-provided prefix to form the complete role name. Note: Some services do
 not support the CustomSuffix parameter. If you provide an optional suffix and the
 operation fails, try the operation again without the suffix.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": "^[\\w+=,.@-]{1,64}$|^$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Description": {
          "description": "A meaningful description for the role.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "",
            "pattern": ".*"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "AWSServiceName",
          "CustomSuffix",
          "Description"
        ]
      },
      "required": [
        "AWSServiceName"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
```

```
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2epp05svrlwod

Classifications:

- Deployment | Advanced stack components | KMS key | Create (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create KMS Key (review required)",
  "description": "Request a KMS key by describing key permissions or submitting a key
 policy document.",
  "type": "object",
  "properties": {
    "KeyDescription": {
      "description": "A meaningful description of the KMS key; for example, a
 description that indicates that the KMS key is appropriate for a task. The default
 value is an empty string (no description). Note that the description appears in the
 details for the key in the KMS console. Do not include confidential or sensitive
 information as this field may appear in plain text in CloudTrail logs and other
 output.",
      "type": "string",
      "maxLength": 5000
    },
    "AliasName": {
      "description": "An alias name for the KMS key. The alias name must be unique in
 the AWS account and region, can be up to 256 characters in length, and is limited to
 use characters a-z, A-Z, 0-9, and /_-",
      "type": "string",
      "pattern": "^[a-zA-Z0-9/_-]{1,256}$"
    },
    "KeyRotation": {
      "description": "True if the KMS key should be rotated, false if it should not.
 Default is true.",
      "type": "boolean",
      "default": true
    },
    "KeyPermissions": {
```

```
      "description": "Detailed information about the key permissions, or a key
policy document to be attached to the key (paste the policy document into the value
field).",
      "type": "string",
      "maxLength": 5000
    },
    "MultiRegion": {
      "description": "True to create multi-region key, false to create single-region
key. Default value is false.",
      "type": "boolean",
      "default": false
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
```

```
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "Operation": {
      "description": "Must be Create.",
      "type": "string",
      "default": "Create",
      "enum": [
        "Create"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "KeyDescription",
      "AliasName",
      "KeyRotation",
      "KeyPermissions",
      "MultiRegion",
      "Tags",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "KeyDescription",
    "KeyPermissions",
    "Operation"
  ]
}
```

Schema for Change Type ct-2fqmbyud166z9

Classifications:

- Management | Directory Service | DNS | Update conditional forwarder


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update AD DNS Conditional Forwarder",
  "description": "Update AD DNS conditional forwarder for a remote domain. For multi-
account landing zone (MALZ), use this change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateADDNSConditionalForwarder-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateADDNSConditionalForwarder-Admin"
      ],
      "default": "AWSManagedServices-UpdateADDNSConditionalForwarder-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RemoteDomainName": {
          "description": "The fully qualified domain name (FQDN) of the remote
 domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+[.]?$"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
        "IPAddresses": {
          "description": "A list of private IP addresses of the remote DNS servers
 associated with the conditional forwarder.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(10\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3}))$|^(192\\.168\\.(\
\d{1,3})\\.(\\d{1,3}))$|^(172\\.(1[6-9]|2[0-9]|3[0-1])\\.[0-9]{1,3}\\.[0-9]{1,3})$"
          },
          "minItems": 1,
          "maxItems": 5,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "RemoteDomainName",
          "IPAddresses"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RemoteDomainName",
        "IPAddresses"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2fzh1wckpl7f5

Classifications:

- Management | Managed Firewall | Outbound (Palo Alto) | Delete allow list


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Allow List",
  "description": "Delete an allow list file for AMS managed Palo Alto firewall -
 Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be DeleteAllowList.",
      "type": "string",
      "enum": [
        "DeleteAllowList"
      ],
      "default": "DeleteAllowList"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AllowListName": {
          "description": "The name of the allow list to delete.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AllowListName"
        ]
      },
      "required": [
        "AllowListName"
      ]
    }
  },
  "additionalProperties": false,
```

```
  "metadata": {
    "ui:order": [
      "Parameters",
      "RequestType"
    ]
  },
  "required": [
    "Parameters",
    "RequestType"
  ]
}
```

Schema for Change Type ct-2gd0u847qd9d2

Classifications:

- Deployment | Applications | CodeDeploy deployment group | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create CodeDeploy deployment group",
  "description": "Use to create an AWS CodeDeploy application deployment group, an
 entity that describes what instances to deploy a given application to.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "The reason for the request.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-sp9lrk00000000000",
      "type": "string",
      "enum": [
        "stm-sp9lrk00000000000"
```

```
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
```

```
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "CodeDeployApplicationName": {
          "description": "The name of an AWS CodeDeploy application.",
          "type": "string",
          "minLength": 1,
          "maxLength": 100,
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        },
        "CodeDeployAutoScalingGroups": {
          "description": "The Auto Scaling groups to be updated by AWS CodeDeploy when
new instances are created. Note: Do not associate an Auto Scaling group with more than
one deployment group.",
          "type": "array",
          "items": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255,
            "pattern": "^[a-zA-Z0-9._+=,@-]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 10
        },
        "CodeDeployDeploymentConfigName": {
          "description": "The configuration for deployment operations: as many
instances as possible at once, half of the instances at a time, or only one instance
at a time.",
          "type": "string",
          "enum": [
            "CodeDeployDefault.AllAtOnce",
            "CodeDeployDefault.HalfAtATime",
            "CodeDeployDefault.OneAtATime"
          ],
          "default": "CodeDeployDefault.OneAtATime"
        },
        "CodeDeployDeploymentGroupName": {
          "description": "A name for the deployment group.",
          "type": "string",
          "minLength": 1,
          "maxLength": 100,
          "pattern": "^[a-zA-Z0-9._+=,@-]{1,100}$"
        },
        "CodeDeployServiceRoleArn": {
```

```
          "description": "The Amazon Resource Name (ARN) of an existing CodeDeploy
 service role that grants permission to make calls to AWS services on your behalf, in
 the form arn:aws:iam::ACCOUNT_ID:role/aws-codedeploy-role.",
          "type": "string"
        }
      },
      "additionalProperties": false,
      "required": [
        "CodeDeployApplicationName",
        "CodeDeployAutoScalingGroups",
        "CodeDeployDeploymentGroupName",
        "CodeDeployServiceRoleArn"
      ]
    }
  },
  "additionalProperties": false,
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2ha68tpd7nr3y

Classifications:

- Deployment | Managed landing zone | Application account | Create VPC Additional CIDR and Subnets

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Application Account CIDRs",
  "description": "Create an additional VPC CIDR, or subnets, or both, for an existing
 application account VPC. Add up to five public and twenty private subnet tiers to the
 additional CIDR, or to existing CIDRs under the VPC. A subnet tier is a set of subnets
 provisioned in two or three Availability Zones (AZ).",
  "type": "object",
```

```
  "properties": {
    "VPCId": {
      "description": "The ID of the VPC to add additional CIDRs or subnets to.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VPCCIDR": {
          "description": "The Classless Inter-Domain Routing (CIDR) range to be added
 to the existing application account VPC.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "RouteType": {
          "description": "The AWS Transit Gateway application route table connection
 type. For this VPC extension to accept connections from other VPCs, use routable. For
 it to not accept those connections, use isolated. The default is routable.",
          "type": "string",
          "enum": [
            "isolated",
            "routable"
          ],
          "default": "routable"
        },
        "PrivateRouteTableAZ1ID": {
          "description": "The route table ID for the private subnets in AZ1.",
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PrivateRouteTableAZ2ID": {
          "description": "The route table ID for the private subnets in AZ2.",
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PrivateRouteTableAZ3ID": {
          "description": "The route table ID for the private subnets in AZ3.",
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PublicRouteTableAZ1ID": {
          "description": "The route table ID for the public subnets in AZ1.",
```

```
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PublicRouteTableAZ2ID": {
          "description": "The route table ID for the public subnets in AZ2.",
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PublicRouteTableAZ3ID": {
          "description": "The route table ID for the public subnets in AZ3.",
          "type": "string",
          "pattern": "^rtb-([a-z0-9]{8}|[a-z0-9]{17})|^$"
        },
        "PublicSubnet1AZ1CIDR": {
          "description": "The CIDR for the first public subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet1AZ2CIDR": {
          "description": "The CIDR for the first public subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet1AZ3CIDR": {
          "description": "The CIDR for the first public subnet tier in AZ3. Use only if
 three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet2AZ1CIDR": {
          "description": "The CIDR for the second public subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet2AZ2CIDR": {
          "description": "The CIDR for the second public subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
```

```
        "PublicSubnet2AZ3CIDR": {
          "description": "The CIDR for the second public subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet3AZ1CIDR": {
          "description": "The CIDR for the third public subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet3AZ2CIDR": {
          "description": "The CIDR for the third public subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet3AZ3CIDR": {
          "description": "The CIDR for the third public subnet tier in AZ3. Use only if
 three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet4AZ1CIDR": {
          "description": "The CIDR for the fourth public subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet4AZ2CIDR": {
          "description": "The CIDR for the fourth public subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet4AZ3CIDR": {
          "description": "The CIDR for the fourth public subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
```

```
        },
        "PublicSubnet5AZ1CIDR": {
          "description": "The CIDR for the fifth public subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet5AZ2CIDR": {
          "description": "The CIDR for the fifth public subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PublicSubnet5AZ3CIDR": {
          "description": "The CIDR for the fifth public subnet tier in AZ3. Use only if
 three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet1AZ1CIDR": {
          "description": "The CIDR for the first private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet1AZ2CIDR": {
          "description": "The CIDR for the first private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet1AZ3CIDR": {
          "description": "The CIDR for the first private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet2AZ1CIDR": {
          "description": "The CIDR for the second private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
```

```
        },
        "PrivateSubnet2AZ2CIDR": {
          "description": "The CIDR for the second private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet2AZ3CIDR": {
          "description": "The CIDR for the second private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet3AZ1CIDR": {
          "description": "The CIDR for the third private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet3AZ2CIDR": {
          "description": "The CIDR for the third private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet3AZ3CIDR": {
          "description": "The CIDR for the third private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet4AZ1CIDR": {
          "description": "The CIDR for the fourth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet4AZ2CIDR": {
          "description": "The CIDR for the fourth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
```

```
        },
        "PrivateSubnet4AZ3CIDR": {
          "description": "The CIDR for the fourth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet5AZ1CIDR": {
          "description": "The CIDR for the fifth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet5AZ2CIDR": {
          "description": "The CIDR for the fifth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet5AZ3CIDR": {
          "description": "The CIDR for the fifth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet6AZ1CIDR": {
          "description": "The CIDR for the sixth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet6AZ2CIDR": {
          "description": "The CIDR for the sixth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet6AZ3CIDR": {
          "description": "The CIDR for the sixth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
```

```
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet7AZ1CIDR": {
          "description": "The CIDR for the seventh private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet7AZ2CIDR": {
          "description": "The CIDR for the seventh private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet7AZ3CIDR": {
          "description": "The CIDR for the seventh private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet8AZ1CIDR": {
          "description": "The CIDR for the eighth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet8AZ2CIDR": {
          "description": "The CIDR for the eighth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet8AZ3CIDR": {
          "description": "The CIDR for the eighth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet9AZ1CIDR": {
          "description": "The CIDR for the ninth private subnet tier in AZ1.",
          "type": "string",
```

```
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet9AZ2CIDR": {
          "description": "The CIDR for the ninth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet9AZ3CIDR": {
          "description": "The CIDR for the ninth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet10AZ1CIDR": {
          "description": "The CIDR for the tenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet10AZ2CIDR": {
          "description": "The CIDR for the tenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet10AZ3CIDR": {
          "description": "The CIDR for the tenth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet11AZ1CIDR": {
          "description": "The CIDR for the eleventh private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet11AZ2CIDR": {
          "description": "The CIDR for the eleventh private subnet tier in AZ2.",
          "type": "string",
```

```
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet11AZ3CIDR": {
          "description": "The CIDR for the eleventh private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet12AZ1CIDR": {
          "description": "The CIDR for the twelfth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet12AZ2CIDR": {
          "description": "The CIDR for the twelfth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet12AZ3CIDR": {
          "description": "The CIDR for the twelfth private subnet tier in AZ3. Use only
 if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet13AZ1CIDR": {
          "description": "The CIDR for the thirteenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet13AZ2CIDR": {
          "description": "The CIDR for the thirteenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet13AZ3CIDR": {
          "description": "The CIDR for the thirteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
```

```
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet14AZ1CIDR": {
          "description": "The CIDR for the fourteenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet14AZ2CIDR": {
          "description": "The CIDR for the fourteenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet14AZ3CIDR": {
          "description": "The CIDR for the fourteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet15AZ1CIDR": {
          "description": "The CIDR for the fifteenth private subnet tier in AZ31.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet15AZ2CIDR": {
          "description": "The CIDR for the fifteenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet15AZ3CIDR": {
          "description": "The CIDR for the fifteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet16AZ1CIDR": {
          "description": "The CIDR for the sixteenth private subnet tier in AZ1.",
```

```
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet16AZ2CIDR": {
          "description": "The CIDR for the sixteenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet16AZ3CIDR": {
          "description": "The CIDR for the sixteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet17AZ1CIDR": {
          "description": "The CIDR for the seventeenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet17AZ2CIDR": {
          "description": "The CIDR for the seventeenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet17AZ3CIDR": {
          "description": "The CIDR for the seventeenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet18AZ1CIDR": {
          "description": "The CIDR for the eighteenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet18AZ2CIDR": {
          "description": "The CIDR for the eighteenth private subnet tier in AZ2.",
```

```
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet18AZ3CIDR": {
          "description": "The CIDR for the eighteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet19AZ1CIDR": {
          "description": "The CIDR for the ninteenth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet19AZ2CIDR": {
          "description": "The CIDR for the ninteenth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet19AZ3CIDR": {
          "description": "The CIDR for the ninteenth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet20AZ1CIDR": {
          "description": "The CIDR for the twentieth private subnet tier in AZ1.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet20AZ2CIDR": {
          "description": "The CIDR for the twentieth private subnet tier in AZ2.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        },
        "PrivateSubnet20AZ3CIDR": {
```

```
          "description": "The CIDR for the twentieth private subnet tier in AZ3. Use
 only if three AZs are chosen.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "VPCCIDR",
          "RouteType",
          "PrivateRouteTableAZ1ID",
          "PrivateRouteTableAZ2ID",
          "PrivateRouteTableAZ3ID",
          "PublicRouteTableAZ1ID",
          "PublicRouteTableAZ2ID",
          "PublicRouteTableAZ3ID",
          "PublicSubnet1AZ1CIDR",
          "PublicSubnet1AZ2CIDR",
          "PublicSubnet1AZ3CIDR",
          "PublicSubnet2AZ1CIDR",
          "PublicSubnet2AZ2CIDR",
          "PublicSubnet2AZ3CIDR",
          "PublicSubnet3AZ1CIDR",
          "PublicSubnet3AZ2CIDR",
          "PublicSubnet3AZ3CIDR",
          "PublicSubnet4AZ1CIDR",
          "PublicSubnet4AZ2CIDR",
          "PublicSubnet4AZ3CIDR",
          "PublicSubnet5AZ1CIDR",
          "PublicSubnet5AZ2CIDR",
          "PublicSubnet5AZ3CIDR",
          "PrivateSubnet1AZ1CIDR",
          "PrivateSubnet1AZ2CIDR",
          "PrivateSubnet1AZ3CIDR",
          "PrivateSubnet2AZ1CIDR",
          "PrivateSubnet2AZ2CIDR",
          "PrivateSubnet2AZ3CIDR",
          "PrivateSubnet3AZ1CIDR",
          "PrivateSubnet3AZ2CIDR",
          "PrivateSubnet3AZ3CIDR",
          "PrivateSubnet4AZ1CIDR",
          "PrivateSubnet4AZ2CIDR",
          "PrivateSubnet4AZ3CIDR",
```

```
            "PrivateSubnet5AZ1CIDR",
            "PrivateSubnet5AZ2CIDR",
            "PrivateSubnet5AZ3CIDR",
            "PrivateSubnet6AZ1CIDR",
            "PrivateSubnet6AZ2CIDR",
            "PrivateSubnet6AZ3CIDR",
            "PrivateSubnet7AZ1CIDR",
            "PrivateSubnet7AZ2CIDR",
            "PrivateSubnet7AZ3CIDR",
            "PrivateSubnet8AZ1CIDR",
            "PrivateSubnet8AZ2CIDR",
            "PrivateSubnet8AZ3CIDR",
            "PrivateSubnet9AZ1CIDR",
            "PrivateSubnet9AZ2CIDR",
            "PrivateSubnet9AZ3CIDR",
            "PrivateSubnet10AZ1CIDR",
            "PrivateSubnet10AZ2CIDR",
            "PrivateSubnet10AZ3CIDR",
            "PrivateSubnet11AZ1CIDR",
            "PrivateSubnet11AZ2CIDR",
            "PrivateSubnet11AZ3CIDR",
            "PrivateSubnet12AZ1CIDR",
            "PrivateSubnet12AZ2CIDR",
            "PrivateSubnet12AZ3CIDR",
            "PrivateSubnet13AZ1CIDR",
            "PrivateSubnet13AZ2CIDR",
            "PrivateSubnet13AZ3CIDR",
            "PrivateSubnet14AZ1CIDR",
            "PrivateSubnet14AZ2CIDR",
            "PrivateSubnet14AZ3CIDR",
            "PrivateSubnet15AZ1CIDR",
            "PrivateSubnet15AZ2CIDR",
            "PrivateSubnet15AZ3CIDR",
            "PrivateSubnet16AZ1CIDR",
            "PrivateSubnet16AZ2CIDR",
            "PrivateSubnet16AZ3CIDR",
            "PrivateSubnet17AZ1CIDR",
            "PrivateSubnet17AZ2CIDR",
            "PrivateSubnet17AZ3CIDR",
            "PrivateSubnet18AZ1CIDR",
            "PrivateSubnet18AZ2CIDR",
            "PrivateSubnet18AZ3CIDR",
            "PrivateSubnet19AZ1CIDR",
            "PrivateSubnet19AZ2CIDR",
```

```
          "PrivateSubnet19AZ3CIDR",
          "PrivateSubnet20AZ1CIDR",
          "PrivateSubnet20AZ2CIDR",
          "PrivateSubnet20AZ3CIDR"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VPCId",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "VPCId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2hh93eyzmwbkd

Classifications:

- Management | Advanced stack components | S3 storage | Update versioning

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change S3 Bucket Versioning Setting",
  "description": "Change S3 bucket versioning setting through direct API calls.
 The S3 bucket can be standalone or belong to a CloudFormation stack; in the latter
 case, the change might cause stack drift. To avoid causing stack drift, please use
 ct-1gi93jhvj28eg instead, or ct-361tlo1k7339x if the S3 bucket was provisioned via CFN
 ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateBucketVersioning.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-UpdateBucketVersioning"
      ],
      "default": "AWSManagedServices-UpdateBucketVersioning"
    },
    "Region": {
      "description": "The AWS Region in which the resource is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BucketName": {
          "description": "The name of the bucket to update.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(mc|ams|awsms)-)[a-z0-9][-.a-z0-9]{1,61}[a-z0-9]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Versioning": {
          "description": "Enabled to maintain bucket versioning, Suspended to disable
 bucket versioning. Use S3 Versioning to keep multiple versions of an object in one
 bucket.",
          "type": "string",
          "enum": [
            "Enabled",
            "Suspended"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "BucketName",
          "Versioning"
        ]
      },
      "additionalProperties": false,
      "required": [
        "BucketName",
        "Versioning"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2hhqzgxvkcig8

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create access key

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Access Key",
  "description": "Create a new AWS secret access key and corresponding AWS access key
 ID for the specified user.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateIAMAccessKeyV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateIAMAccessKeyV2"
      ],
      "default": "AWSManagedServices-CreateIAMAccessKeyV2"
    },
    "Region": {
```

```
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "UserARN": {
          "description": "The ARN of the IAM user that the new key will belong to.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:user/[\\w+=,.@-]+$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "UserARN"
```

```
        ]
      },
      "required": [
        "UserARN"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2hhud2lx01tq7

Classifications:

- Management | AWS Backup | Backup job | Start

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start Backup Job",
  "description": "Start an AWS Backup service backup job to create a one-time snapshot
 of the specified resource.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartBackupJob.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartBackupJob"
      ],
      "default": "AWSManagedServices-StartBackupJob"
```

```
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "BackupVaultName": {
          "description": "The name of the target backup vault. The backup vault name is
 case sensitive and must contain from 2 to 50 alphanumeric characters or hyphens. If a
 name is not specified, the name ams-manual-backups is used.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "ams-manual-backups",
            "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$"
          },
          "maxItems": 1
        },
        "CompleteWindowMinutes": {
          "description": "The amount of time AWS Backup attempts a backup
 before canceling the job and returning an error. If a time is specified, then
 StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
 must be at least 60 minutes greater than StartWindowMinutes.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(1[2-8][0-9]|19[0-9]|[2-9][0-9]{2}|[1-8][0-9]{3}|9[0-8][0-9]
{2}|99[0-8][0-9]|999[0-9]|[1-8][0-9]{4}|9[0-8][0-9]{3}|99[0-8][0-9]{2}|999[0-8][0-9]|
9999[0-9]|[1-8][0-9]{5}|9[0-8][0-9]{4}|99[0-8][0-9]{3}|999[0-8][0-9]{2}|9999[0-8][0-9]|
99999[0-9]|[1-8][0-9]{6}|9[0-8][0-9]{5}|99[0-8][0-9]{4}|999[0-8][0-9]{3}|9999[0-8][0-9]
{2}|99999[0-8][0-9]|999999[0-9]|[1-8][0-9]{7}|9[0-8][0-9]{6}|99[0-8][0-9]{5}|999[0-8]
[0-9]{4}|9999[0-8][0-9]{3}|99999[0-8][0-9]{2}|999999[0-8][0-9]|9999999[0-9]|[1-8][0-9]
{8}|9[0-8][0-9]{7}|99[0-8][0-9]{6}|999[0-8][0-9]{5}|9999[0-8][0-9]{4}|99999[0-8][0-9]
{3}|999999[0-8][0-9]{2}|9999999[0-8][0-9]|99999999[0-9]|1[0-9]{9}|20[0-9]{8}|21[0-3]
[0-9]{7}|214[0-6][0-9]{6}|2147[0-3][0-9]{5}|21474[0-7][0-9]{4}|214748[0-2][0-9]{3}|
2147483[0-5][0-9]{2}|21474836[0-3][0-9]|214748364[0-7])$"
          },
          "maxItems": 1
        },
        "DeleteAfterDays": {
```

```
          "description": "The number of days after creation that a backup is deleted.
 Valid values are between 1 and 35600. If a value is not specified, the backup never
 expires.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|[12][0-9]{4}|3[0-4][0-9]{3}|35[0-5]
[0-9]{2}|35600)$"
          },
          "maxItems": 1
        },
        "ResourceArn": {
          "description": "The Amazon Resource Name (ARN) of the AWS resource to
 backup.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:([a-z][a-z0-9-]+):([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        },
        "StartWindowMinutes": {
          "description": "The amount of time in minutes before beginning a backup. The
 minimum value is 60. If a value is not specified, the backup starts immediately.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([6-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|[1-8]
[0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|[1-8][0-9]{4}|9[0-8][0-9]{3}|99[0-8][0-9]
{2}|999[0-8][0-9]|9999[0-9]|[1-8][0-9]{5}|9[0-8][0-9]{4}|99[0-8][0-9]{3}|999[0-8][0-9]
{2}|9999[0-8][0-9]|99999[0-9]|[1-8][0-9]{6}|9[0-8][0-9]{5}|99[0-8][0-9]{4}|999[0-8]
[0-9]{3}|9999[0-8][0-9]{2}|99999[0-8][0-9]|999999[0-9]|[1-8][0-9]{7}|9[0-8][0-9]{6}|
99[0-8][0-9]{5}|999[0-8][0-9]{4}|9999[0-8][0-9]{3}|99999[0-8][0-9]{2}|999999[0-8][0-9]|
9999999[0-9]|[1-8][0-9]{8}|9[0-8][0-9]{7}|99[0-8][0-9]{6}|999[0-8][0-9]{5}|9999[0-8]
[0-9]{4}|99999[0-8][0-9]{3}|999999[0-8][0-9]{2}|9999999[0-8][0-9]|99999999[0-9]|1[0-9]
{9}|20[0-9]{8}|21[0-3][0-9]{7}|214[0-6][0-9]{6}|2147[0-3][0-9]{5}|21474[0-7][0-9]{4}|
214748[0-2][0-9]{3}|2147483[0-5][0-9]{2}|21474836[0-3][0-9]|214748364[0-7])$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
```

```
        "ui:order": [
          "BackupVaultName",
          "CompleteWindowMinutes",
          "DeleteAfterDays",
          "ResourceArn",
          "StartWindowMinutes"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResourceArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2hxcllf1b4ey0

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint (MongoDB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS source endpoint for MongoDB",
  "description": "Use to create a Database Migration Service (DMS) source endpoint for
 MongoDB.",
```

```
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
           "minLength": 1,
           "maxLength": 127
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Key",
```

```
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-pud4ghhkp7395n9bc.",
      "type": "string",
      "enum": [
        "stm-pud4ghhkp7395n9bc"
      ],
      "default": "stm-pud4ghhkp7395n9bc"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) for the certificate to use
with the source. This is required if SslMode = verify-full.",
          "pattern": "^$|^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:cert:[A-Z0-9]+$"
        },
        "DatabaseName": {
          "type": "string",
          "description": "The name of the source database."
        },
        "EndpointIdentifier": {
          "type": "string",
```

```
        "description": "A meaningful identifier for the source database endpoint.
Must be unique for all endpoints owned by your AWS account in the current region. Must
begin with a letter, must contain only ASCII letters, digits and hyphens and must not
end with a hyphen or contain two consecutive hyphens.",
        "pattern": "^$|(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$",
        "default": ""
      },
      "EngineName": {
        "type": "string",
        "description": "Must be mongodb.",
        "enum": [
          "mongodb"
        ]
      },
      "ExtraConnectionAttributes": {
        "type": "string",
        "description": "Additional attributes associated with the connection. See AWS
documentation for more information on the supported extra connection attributes for
MongoDb."
      },
      "Password": {
        "type": "string",
        "description": "The password to be used to log in to the source database.
Leave blank if MongoDbAuthType = no.",
        "metadata": {
          "ams:sensitive": true
        }
      },
      "Port": {
        "type": "integer",
        "description": "The port used by the source database.",
        "minimum": 1,
        "maximum": 65535
      },
      "ServerName": {
        "type": "string",
        "description": "The name of the server where the source database resides."
      },
      "SslMode": {
        "type": "string",
        "description": "The SSL mode to use for the SSL connection.",
        "enum": [
          "none",
          "require",
```

```
            "verify-full"
          ],
          "default": "none"
        },
        "Username": {
          "type": "string",
          "description": "The user name to be used to log in to the source database.
Leave blank if MongoDbAuthType = no.",
          "metadata": {
            "ams:sensitive": true
          }
        },
        "MongoDbAuthMechanism": {
          "type": "string",
          "description": "The authentication mechanism used to access the MongoDB
source endpoint. Do not use if MongoDbAuthType = no.",
          "enum": [
            "default",
            "mongodb_cr",
            "scram_sha_1"
          ],
          "default": "default"
        },
        "MongoDbAuthSource": {
          "type": "string",
          "description": "The MongoDB database name. Do not use if MongoDbAuthType =
no.",
          "default": "admin"
        },
        "MongoDbAuthType": {
          "type": "string",
          "description": "The authentication type or mode used to access the MongoDB
source endpoint.",
          "enum": [
            "no",
            "password"
          ],
          "default": "no"
        },
        "MongoDbDocsToInvestigate": {
          "type": "string",
          "description": "The number of documents to preview to determine the document
organization. Use if MongoDbMetadataMode = one. Must be a positive value greater than
0.",
```

```
          "pattern": "^[1-9]{1}$|^[1-9]{1}[0-9]+$",
          "default": "1000"
        },
        "MongoDbExtractDocId": {
          "type": "string",
          "description": "True to extract the MongoDB document ID as a separate column;
false to not. Use if MongoDbMetadataMode = none.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "MongoDbMetadataMode": {
          "type": "string",
          "description": "The mode used for MongoDB metadata. For document mode use
none, for table mode use one.",
          "enum": [
            "none",
            "one"
          ],
          "default": "none"
        }
      },
      "metadata": {
        "ui:order": [
          "EndpointIdentifier",
          "EngineName",
          "ServerName",
          "Port",
          "DatabaseName",
          "Username",
          "Password",
          "SslMode",
          "CertificateArn",
          "ExtraConnectionAttributes",
          "MongoDbAuthType",
          "MongoDbAuthMechanism",
          "MongoDbAuthSource",
          "MongoDbMetadataMode",
          "MongoDbDocsToInvestigate",
          "MongoDbExtractDocId"
        ]
      },
```

```
      "required": [
        "EngineName",
        "ServerName",
        "Port",
        "DatabaseName"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2hyozbpa0sx0m

Classifications:

- Deployment | AWS Backup | Backup plan | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create AWS Backup Plan",
  "description": "Create an AWS Backup plan, a policy expression that defines when and
 how you want to back up your AWS resources.",
```

```
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the backup plan to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "This parameter is deprecated and will be removed in the future.
AMS generates a unique, random, name for the resource and that becomes the StackName
in the AMS console.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "StackTemplateId": {
     "description": "Must be stm-sc68a620000000000",
     "type": "string",
     "enum": [
       "stm-sc68a620000000000"
     ],
     "default": "stm-sc68a620000000000"
   },
   "TimeoutInMinutes": {
     "description": "The maximum amount of time, in minutes, to allow for the
execution of the change. This does not prolong execution, but the RFC fails if the
change is not completed within the specified time.",
     "type": "number",
     "minimum": 0,
     "maximum": 360,
     "default": 60
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "BackupPlanName": {
         "type": "string",
```

```
          "description": "A meaningful name for the AWS Backup plan."
        },
        "ResourceTagKey": {
          "type": "string",
          "description": "The tag key (case sensitive) of the resources to be backed
up. For example, if you want to use a tag key:value pair like 'Department:accounting',
you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
ResourceTagValue."
        },
        "ResourceTagValue": {
          "type": "string",
          "description": "The tag value (case sensitive) of the resources to be backed
up. For example, if you want to use a tag key:value pair like 'Department:accounting',
you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
ResourceTagValue."
        },
        "WindowsVSS": {
          "type": "string",
          "description": "Enabled to use the Windows Volume Shadow Copy Service
(VSS) backup option in AWS Backup. Disabled to create a regular backup. Default is
disabled.",
          "enum": [
            "disabled",
            "enabled"
          ],
          "default": "disabled"
        },
        "BackupRule1Name": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan rule #1.",
          "default": "BackupRule1"
        },
        "BackupRule1Vault": {
          "type": "string",
          "description": "The name of the AWS Backup Vault to be used in the AWS Backup
plan rule #1.",
          "default": "ams-custom-backups"
        },
        "BackupRule1CompletionWindowMinutes": {
          "type": "integer",
          "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
```

```
          "minimum": 1,
          "maximum": 99000,
          "default": 1400
        },
        "BackupRule1ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule1DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule1MoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule1StartWindowMinutes": {
          "type": "integer",
          "description": "The period of time, in minutes, after a backup is scheduled
to wait before a job is canceled if it doesn't start successfully.",
          "minimum": 60,
          "maximum": 99000,
          "default": 180
        },
        "BackupRule1RecoveryPointTagKey": {
          "type": "string",
          "description": "A key for the tag that is assigned to all created recovery
points for backup rule #1.",
          "default": ""
        },
        "BackupRule1RecoveryPointTagValue": {
```

```
          "type": "string",
          "description": "A value for the BackupRule1RecoveryPointTagKey.",
          "default": ""
        },
        "BackupRule1EnableContinuousBackup": {
          "type": "string",
          "description": "True to create a continuous backup rule, false to not create
 the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule1CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #1: The Amazon Resource Name (ARN) of
 the destination backup vault for the copied backup.",
          "default": "",
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule1CAMoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #1 copy actions: The number of days
 after creation before the recovery point is moved to cold storage. Valid values are
 between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
 Only Amazon EFS file system backups can be transitioned to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule1CopyActionsDeleteAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #1 copy actions: The number of days
 after creation that a recovery point is deleted. Valid values are between 1 and 35600.
 If a value is not specified, the backup never expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule2Name": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan rule #2.",
```

```
          "default": ""
        },
        "BackupRule2Vault": {
          "type": "string",
          "description": "The name of the AWS Backup Vault to be used in the AWS Backup
plan rule #2.",
          "default": "ams-custom-backups"
        },
        "BackupRule2CompletionWindowMinutes": {
          "type": "integer",
          "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
          "minimum": 1,
          "maximum": 99000,
          "default": 1400
        },
        "BackupRule2ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule2DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule2MoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule2StartWindowMinutes": {
```

```
          "type": "integer",
          "description": "The period of time, in minutes, after a backup is scheduled
 to wait before a job is canceled if it doesn't start successfully.",
          "minimum": 60,
          "maximum": 99000,
          "default": 180
        },
        "BackupRule2RecoveryPointTagKey": {
          "type": "string",
          "description": "A key for the tag that is assigned to all created recovery
 points for backup rule #2."
        },
        "BackupRule2RecoveryPointTagValue": {
          "type": "string",
          "description": "A value for the BackupRule2RecoveryPointTagKey."
        },
        "BackupRule2EnableContinuousBackup": {
          "type": "string",
          "description": "True to create a continuous backup rule, false to not create
 the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule2CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #2: The Amazon Resource Name (ARN) of
 the destination backup vault for the copied backup.",
          "default": "",
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule2CAMoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #2 copy actions: The number of days
 after creation before the recovery point is moved to cold storage. Valid values are
 between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
 Only Amazon EFS file system backups can be transitioned to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
```

```
      "BackupRule2CopyActionsDeleteAfterDays": {
        "type": "integer",
        "description": "For backup plan rule #2 copy actions: The number of days
after creation that a recovery point is deleted. Valid values are between 1 and 35600.
If a value is not specified, the backup never expires.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule3Name": {
        "type": "string",
        "description": "A meaningful name for the AWS Backup plan rule #3.",
        "default": ""
      },
      "BackupRule3Vault": {
        "type": "string",
        "description": "The name of the AWS Backup Vault to be used in the AWS Backup
plan rule #3.",
        "default": "ams-custom-backups"
      },
      "BackupRule3CompletionWindowMinutes": {
        "type": "integer",
        "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
        "minimum": 1,
        "maximum": 99000,
        "default": 1400
      },
      "BackupRule3ScheduleExpression": {
        "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
        "type": "string",
        "pattern": "^(cron|rate)\\(.*\\)$"
      },
      "BackupRule3DeleteAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
        "minimum": 0,
        "maximum": 35600,
```

```
          "default": 0
        },
        "BackupRule3MoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule3StartWindowMinutes": {
          "type": "integer",
          "description": "The period of time, in minutes, after a backup is scheduled
to wait before a job is canceled if it doesn't start successfully.",
          "minimum": 60,
          "maximum": 99000,
          "default": 180
        },
        "BackupRule3RecoveryPointTagKey": {
          "type": "string",
          "description": "A key for the tag that is assigned to all created recovery
points for backup rule #3."
        },
        "BackupRule3RecoveryPointTagValue": {
          "type": "string",
          "description": "A value for the BackupRule3RecoveryPointTagKey."
        },
        "BackupRule3EnableContinuousBackup": {
          "type": "string",
          "description": "True to create a continuous backup rule, false to not create
the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule3CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #3: The Amazon Resource Name (ARN) of
the destination backup vault for the copied backup.",
          "default": "",
```

```
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule3CAMoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #3 copy actions: The number of days
 after creation before the recovery point is moved to cold storage. Valid values are
 between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
 Only Amazon EFS file system backups can be transitioned to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule3CopyActionsDeleteAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #3 copy actions: The number of days
 after creation that a recovery point is deleted. Valid values are between 1 and 35600.
 If a value is not specified, the backup never expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule4Name": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan rule #4.",
          "default": ""
        },
        "BackupRule4Vault": {
          "type": "string",
          "description": "The name of the AWS Backup Vault to be used in the AWS Backup
 plan rule #4.",
          "default": "ams-custom-backups"
        },
        "BackupRule4CompletionWindowMinutes": {
          "type": "integer",
          "description": "The amount of time, in minutes, that AWS Backup attempts a
 backup before canceling the job and returning an error. If a time is specified, then
 StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
 must be at least 60 minutes greater than StartWindowMinutes.",
          "minimum": 1,
          "maximum": 99000,
          "default": 1400
        },
        "BackupRule4ScheduleExpression": {
```

```
        "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
        "type": "string",
        "pattern": "^(cron|rate)\\(.*\\)$"
      },
      "BackupRule4DeleteAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule4MoveToColdStorageAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that a backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule4StartWindowMinutes": {
        "type": "integer",
        "description": "The period of time, in minutes, after a backup is scheduled
to wait before a job is canceled if it doesn't start successfully.",
        "minimum": 60,
        "maximum": 99000,
        "default": 180
      },
      "BackupRule4RecoveryPointTagKey": {
        "type": "string",
        "description": "A key for the tag that is assigned to all created recovery
points for backup rule #4."
      },
      "BackupRule4RecoveryPointTagValue": {
        "type": "string",
        "description": "A value for the BackupRule4RecoveryPointTagKey."
      },
      "BackupRule4EnableContinuousBackup": {
        "type": "string",
```

```
          "description": "True to create a continuous backup rule, false to not create
 the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule4CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #4: The Amazon Resource Name (ARN) of
 the destination backup vault for the copied backup.",
          "default": "",
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule4CAMoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #4 copy actions: The number of days
 after creation before the recovery point is moved to cold storage. Valid values are
 between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
 Only Amazon EFS file system backups can be transitioned to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule4CopyActionsDeleteAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #4 copy actions: The number of days
 after creation that a recovery point is deleted. Valid values are between 1 and 35600.
 If a value is not specified, the backup never expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule5Name": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan rule #5.",
          "default": ""
        },
        "BackupRule5Vault": {
          "type": "string",
          "description": "The name of the AWS Backup Vault to be used in the AWS Backup
 plan rule #5.",
```

```
          "default": "ams-custom-backups"
        },
        "BackupRule5CompletionWindowMinutes": {
          "type": "integer",
          "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
          "minimum": 1,
          "maximum": 99000,
          "default": 1400
        },
        "BackupRule5ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule5DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule5MoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule5StartWindowMinutes": {
          "type": "integer",
          "description": "The period of time, in minutes, after a backup is scheduled
to wait before a job is canceled if it doesn't start successfully.",
          "minimum": 60,
          "maximum": 99000,
          "default": 180
```

```
        },
        "BackupRule5RecoveryPointTagKey": {
          "type": "string",
          "description": "A key for the tag that is assigned to all created recovery
 points for backup rule #5."
        },
        "BackupRule5RecoveryPointTagValue": {
          "type": "string",
          "description": "A value for the BackupRule5RecoveryPointTagKey."
        },
        "BackupRule5EnableContinuousBackup": {
          "type": "string",
          "description": "True to create a continuous backup rule, false to not create
 the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule5CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #5: The Amazon Resource Name (ARN) of
 the destination backup vault for the copied backup.",
          "default": "",
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule5CAMoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #5 copy actions: The number of days
 after creation before the recovery point is moved to cold storage. Valid values are
 between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
 Only Amazon EFS file system backups can be transitioned to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule5CopyActionsDeleteAfterDays": {
          "type": "integer",
          "description": "For backup plan rule #5 copy actions: The number of days
 after creation that a recovery point is deleted. Valid values are between 1 and 35600.
 If a value is not specified, the backup never expires.",
          "minimum": 0,
```

```
          "maximum": 35600,
          "default": 0
        },
        "BackupRule6Name": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan rule #6.",
          "default": ""
        },
        "BackupRule6Vault": {
          "type": "string",
          "description": "The name of the AWS Backup Vault to be used in the AWS Backup
plan rule #6.",
          "default": "ams-custom-backups"
        },
        "BackupRule6CompletionWindowMinutes": {
          "type": "integer",
          "description": "The amount of time, in minutes, that AWS Backup attempts a
backup before canceling the job and returning an error. If a time is specified, then
StartWindowMinutes must be specified, and the specified CompleteWindowMinutes time
must be at least 60 minutes greater than StartWindowMinutes.",
          "minimum": 1,
          "maximum": 99000,
          "default": 1400
        },
        "BackupRule6ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule6DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that a backup is deleted.
Valid values are between 1 and 35600. If a value is not specified, the backup never
expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule6MoveToColdStorageAfterDays": {
          "type": "integer",
```

```
          "description": "The number of days after creation that a backup is moved
 to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
 backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule6StartWindowMinutes": {
          "type": "integer",
          "description": "The period of time, in minutes, after a backup is scheduled
 to wait before a job is canceled if it doesn't start successfully.",
          "minimum": 60,
          "maximum": 99000,
          "default": 180
        },
        "BackupRule6RecoveryPointTagKey": {
          "type": "string",
          "description": "A key for the tag that is assigned to all created recovery
 points for backup rule #6."
        },
        "BackupRule6RecoveryPointTagValue": {
          "type": "string",
          "description": "A value for the BackupRule6RecoveryPointTagKey."
        },
        "BackupRule6EnableContinuousBackup": {
          "type": "string",
          "description": "True to create a continuous backup rule, false to not create
 the rule. Default is false.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "BackupRule6CopyActionsDestVaultArn": {
          "type": "string",
          "description": "For backup plan rule #6: The Amazon Resource Name (ARN) of
 the destination backup vault for the copied backup.",
          "default": "",
          "pattern": "^$|^(arn:(aws|aws-cn|aws-us-gov):backup:([a-z]{2}((-gov))?-[a-
z]+-[0-9]){0,1}:[0-9]{12}:backup-vault:[a-zA-Z0-9\\_\\-]+)$"
        },
        "BackupRule6CAMoveToColdStorageAfterDays": {
          "type": "integer",
```

```
        "description": "For backup plan rule #6 copy actions: The number of days
after creation before the recovery point is moved to cold storage. Valid values are
between 1 and 35600. If the value is set to 0, the backup never moves to cold storage.
Only Amazon EFS file system backups can be transitioned to cold storage.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule6CopyActionsDeleteAfterDays": {
        "type": "integer",
        "description": "For backup plan rule #6 copy actions: The number of days
after creation that a recovery point is deleted. Valid values are between 1 and 35600.
If a value is not specified, the backup never expires.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      }
    },
    "metadata": {
      "ui:order": [
        "BackupPlanName",
        "ResourceTagKey",
        "ResourceTagValue",
        "WindowsVSS",
        "BackupRule1Name",
        "BackupRule1Vault",
        "BackupRule1CompletionWindowMinutes",
        "BackupRule1ScheduleExpression",
        "BackupRule1DeleteAfterDays",
        "BackupRule1MoveToColdStorageAfterDays",
        "BackupRule1StartWindowMinutes",
        "BackupRule1RecoveryPointTagKey",
        "BackupRule1RecoveryPointTagValue",
        "BackupRule1EnableContinuousBackup",
        "BackupRule1CopyActionsDestVaultArn",
        "BackupRule1CAMoveToColdStorageAfterDays",
        "BackupRule1CopyActionsDeleteAfterDays",
        "BackupRule2Name",
        "BackupRule2Vault",
        "BackupRule2CompletionWindowMinutes",
        "BackupRule2ScheduleExpression",
        "BackupRule2DeleteAfterDays",
        "BackupRule2MoveToColdStorageAfterDays",
        "BackupRule2StartWindowMinutes",
```

```
            "BackupRule2RecoveryPointTagKey",
            "BackupRule2RecoveryPointTagValue",
            "BackupRule2EnableContinuousBackup",
            "BackupRule2CopyActionsDestVaultArn",
            "BackupRule2CAMoveToColdStorageAfterDays",
            "BackupRule2CopyActionsDeleteAfterDays",
            "BackupRule3Name",
            "BackupRule3Vault",
            "BackupRule3CompletionWindowMinutes",
            "BackupRule3ScheduleExpression",
            "BackupRule3DeleteAfterDays",
            "BackupRule3MoveToColdStorageAfterDays",
            "BackupRule3StartWindowMinutes",
            "BackupRule3RecoveryPointTagKey",
            "BackupRule3RecoveryPointTagValue",
            "BackupRule3EnableContinuousBackup",
            "BackupRule3CopyActionsDestVaultArn",
            "BackupRule3CAMoveToColdStorageAfterDays",
            "BackupRule3CopyActionsDeleteAfterDays",
            "BackupRule4Name",
            "BackupRule4Vault",
            "BackupRule4CompletionWindowMinutes",
            "BackupRule4ScheduleExpression",
            "BackupRule4DeleteAfterDays",
            "BackupRule4MoveToColdStorageAfterDays",
            "BackupRule4StartWindowMinutes",
            "BackupRule4RecoveryPointTagKey",
            "BackupRule4RecoveryPointTagValue",
            "BackupRule4EnableContinuousBackup",
            "BackupRule4CopyActionsDestVaultArn",
            "BackupRule4CAMoveToColdStorageAfterDays",
            "BackupRule4CopyActionsDeleteAfterDays",
            "BackupRule5Name",
            "BackupRule5Vault",
            "BackupRule5CompletionWindowMinutes",
            "BackupRule5ScheduleExpression",
            "BackupRule5DeleteAfterDays",
            "BackupRule5MoveToColdStorageAfterDays",
            "BackupRule5StartWindowMinutes",
            "BackupRule5RecoveryPointTagKey",
            "BackupRule5RecoveryPointTagValue",
            "BackupRule5EnableContinuousBackup",
            "BackupRule5CopyActionsDestVaultArn",
            "BackupRule5CAMoveToColdStorageAfterDays",
```

```
          "BackupRule5CopyActionsDeleteAfterDays",
          "BackupRule6Name",
          "BackupRule6Vault",
          "BackupRule6CompletionWindowMinutes",
          "BackupRule6ScheduleExpression",
          "BackupRule6DeleteAfterDays",
          "BackupRule6MoveToColdStorageAfterDays",
          "BackupRule6StartWindowMinutes",
          "BackupRule6RecoveryPointTagKey",
          "BackupRule6RecoveryPointTagValue",
          "BackupRule6EnableContinuousBackup",
          "BackupRule6CopyActionsDestVaultArn",
          "BackupRule6CAMoveToColdStorageAfterDays",
          "BackupRule6CopyActionsDeleteAfterDays"
        ]
      },
      "required": [
        "BackupPlanName",
        "ResourceTagKey",
        "ResourceTagValue",
        "BackupRule1Name",
        "BackupRule1Vault",
        "BackupRule1ScheduleExpression"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "VpcId",
      "Description",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId"
    ]
  },
  "required": [
    "Name",
    "VpcId",
    "Description",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
```

```
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2j7q1hgf26x5c

Classifications:

- Deployment | Managed landing zone | Management account | Create tools account (with VPC)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Tools Account With VPC",
  "description": "Create a managed AWS landing zone tools account and a VPC with a
 private subnet, an isolated private subnet, and a public subnet. Optionally, also
 create an AWS Backup plan with up to four different rules. Managed AWS landing zone
 core accounts must already be onboarded to AWS Managed Services (AMS).",
  "type": "object",
  "properties": {
    "AccountName": {
      "description": "A name for the new tools account. Maximum length 50 characters.
 The underscore ( _ ) is not allowed.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9]{1}[a-zA-Z0-9.-]{0,49}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AccountEmail": {
          "description": "The email address for the new tools account. The email must
 be unique per account, since it will be used, with your password, to sign in as root
 user to your account. This email address is not used for communication.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
        },
        "ApplicationOUName": {
          "description": "The name of an existing organizational unit (OU) for this
 tools account, in the form of <application ou name>:<child ou name>. The default value
 is applications:tools.",
          "type": "string",
          "default": "applications:tools"
        },
```

```
        "SupportLevel": {
          "description": "The account's AMS support level, Premium or Plus.",
          "type": "string",
          "enum": [
            "plus",
            "premium"
          ]
        },
        "VpcName": {
          "description": "A meaningful name for the tools account VPC. Must be unique
 within this tools account.",
          "type": "string"
        },
        "VpcCIDR": {
          "description": "The Classless Inter-Domain Routing (CIDR) for the VPC.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "TransitGatewayApplicationRouteTableName": {
          "description": "The existing AWS Transit Gateway route table for this tools
 account VPC. The default is defaultAppRouteDomain. To create a new application route
 table, use the Create Application Route Table change type (ct-1urj94c3hdfu5).",
          "type": "string",
          "default": "defaultAppRouteDomain"
        },
        "PrivateSubnetIsolatedCIDR": {
          "description": "The CIDR range to create the isolated private subnet. There
 is no communication back to on premises network from this subnet.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PrivateSubnetCIDR": {
          "description": "The CIDR range to create the private subnet.",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
        },
        "PublicSubnetCIDR": {
          "description": "The CIDR for the public subnet",
          "type": "string",
          "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|
[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
```

```
        },
        "DirectAlertsEmail": {
          "description": "The email address to receive certain resource-based alerts;
 note the onboarding process will create your SNS subscription. If not specified,
 then you can subscribe later using the Subscribe to DirectCustomerAlerts change type
 (ct-3rcl9u1k017wu).",
          "type": "string",
          "pattern": "^[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]
{0,61}[a-zA-Z0-9])?(?:\\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)*$"
        },
        "SamlMetadataDocumentURL": {
          "description": "The URL that points to the Security Assertion Markup
 Language(SAML) metadata document that is used to enable federated access to the tools
 account. Typically, a pre-signed URL for an Amazon S3 object.",
          "type": "string",
          "pattern": "^https://.+$|^$"
        },
        "BackupPlanName": {
          "type": "string",
          "description": "A meaningful name for the AWS Backup plan, which is a policy
 expression that defines when and how you want to back up your AWS resources.",
          "default": "default-backup-plan"
        },
        "ResourceTagKey": {
          "type": "string",
          "description": "The tag key (case sensitive) of the resources to be backed
 up. For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
          "default": "Backup"
        },
        "ResourceTagValue": {
          "type": "string",
          "description": "The tag value (case sensitive) of the resources to be backed
 up. For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
          "default": "True"
        },
        "BackupRule1ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
 initiates a backup job. For example, cron(0 2 ? *) sets a daily backup for 2am UTC
 time.",
          "type": "string",
```

```
        "pattern": "^(cron|rate)\\(.*\\)$",
        "default": "cron(0 2 ? * * )"
      },
      "BackupRule1DeleteAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that the daily backups are
deleted. Valid values are between 1 and 35600. If the value is set to 0, the backup
never expires.",
        "minimum": 0,
        "maximum": 35600,
        "default": 7
      },
      "BackupRule1MoveToColdStorageAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that daily backup are moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule2ScheduleExpression": {
        "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? *) sets a daily backup for 2am UTC
time.",
        "type": "string",
        "pattern": "^(cron|rate)\\(.*\\)$"
      },
      "BackupRule2DeleteAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that weekly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
        "minimum": 0,
        "maximum": 35600,
        "default": 0
      },
      "BackupRule2MoveToColdStorageAfterDays": {
        "type": "integer",
        "description": "The number of days after creation that weekly backups are
moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
the backup never moves to cold storage.",
        "minimum": 0,
        "maximum": 35600,
```

```
          "default": 0
        },
        "BackupRule3ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? *) sets a daily backup for 2am UTC
time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule3DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that monthly backups are
deleted. Valid values are between 1 and 35600. If the value is set to 0, the backup
never expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule3MoveToColdStorageAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that the monthly backups
are moved to cold storage. Valid values are between 1 and 35600. If the value is set
to 0, the backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule4ScheduleExpression": {
          "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
          "type": "string",
          "pattern": "^(cron|rate)\\(.*\\)$"
        },
        "BackupRule4DeleteAfterDays": {
          "type": "integer",
          "description": "The number of days after creation that the yearly backups
are deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        },
        "BackupRule4MoveToColdStorageAfterDays": {
```

```
          "type": "integer",
          "description": "The number of days after creation that the yearly backups are
 moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
 the backup never moves to cold storage.",
          "minimum": 0,
          "maximum": 35600,
          "default": 0
        }
      },
      "metadata": {
        "ui:order": [
          "AccountEmail",
          "ApplicationOUName",
          "SupportLevel",
          "DirectAlertsEmail",
          "SamlMetadataDocumentURL",
          "VpcName",
          "VpcCIDR",
          "TransitGatewayApplicationRouteTableName",
          "PrivateSubnetIsolatedCIDR",
          "PrivateSubnetCIDR",
          "PublicSubnetCIDR",
          "BackupPlanName",
          "ResourceTagKey",
          "ResourceTagValue",
          "BackupRule1ScheduleExpression",
          "BackupRule1DeleteAfterDays",
          "BackupRule1MoveToColdStorageAfterDays",
          "BackupRule2ScheduleExpression",
          "BackupRule2DeleteAfterDays",
          "BackupRule2MoveToColdStorageAfterDays",
          "BackupRule3ScheduleExpression",
          "BackupRule3DeleteAfterDays",
          "BackupRule3MoveToColdStorageAfterDays",
          "BackupRule4ScheduleExpression",
          "BackupRule4DeleteAfterDays",
          "BackupRule4MoveToColdStorageAfterDays"
        ]
      },
      "additionalProperties": false,
      "required": [
        "AccountEmail",
        "SupportLevel",
        "VpcName",
```

```
        "VpcCIDR",
        "PrivateSubnetIsolatedCIDR",
        "PrivateSubnetCIDR",
        "PublicSubnetCIDR",
        "BackupPlanName",
        "ResourceTagKey",
        "ResourceTagValue",
        "BackupRule1ScheduleExpression"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "AccountName",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "AccountName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2jndrh7uit8uf

Classifications:

- Deployment | AMS patterns | Solution | Deploy (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Deploy AMS Patterns",
  "description": "Deploy an AMS pattern to the current account. Patterns provide
 tools, architectures, and step-by-step guidance for implementing the methodologies
 for the migration strategy. Multi-account landing zone accounts can also specify
 OrganizationalUnit to deploy the pattern to all the accounts in that OU.",
  "type": "object",
  "properties": {
    "PatternName": {
      "description": "The name of the AMS pattern to be deployed. Please reach out to
 your AMS Cloud Architect for more details about each pattern before deploying.",
```

```
      "type": "string",
      "enum": [
        "amsAviatrixSALZ",
        "amsAzureADFederationUser",
        "amsCheckAndEnableLDAPSignAndSeal",
        "amsCheckAndRepairSecureChannel",
        "amsCICDwithAwsCodeSuite",
        "amsCISHardening",
        "amscloudcustodianpipeline",
        "amsCloudWatchAlarmScheduler",
        "amsCloudWatchLogGroupsPeriodicRetention",
        "amsCloudwatchLogsRetention",
        "amsControlTowerAccountNotify",
        "amsCrossAccountSnapshotCopier",
        "amsCrowdStrikeAgentManagement",
        "amsCUDOS",
        "amsCWAlarmforDirectConnect",
        "amsCWCustomMetrics",
        "amsCWLogsAgentManagement",
        "amsCWLogsAggregationToSplunk",
        "amsCyberArkIntegration",
        "amsDataSyncMonitor",
        "amsDCMasking",
        "amsDeleteNATGateways",
        "amsDetectAndRemediateVpnAlarms",
        "amsDiskUsageAutomation",
        "amsDotNetPatchesExclusion",
        "amsDR",
        "amsDSMlogsToS3snsLambdaStreaming",
        "amsEBSSnapshotDeletion",
        "amsEbsVolumeSnapshotTagger",
        "amsEnhancedLinuxAccessManagement",
        "amsEOSInstanceChecker",
        "amsEPSEventSNSNotification",
        "amsEventsToSplunk",
        "amsGuardDutySnsIntegration",
        "amsImdsv1ToImdsv2DashboardMonitoringAccount",
        "amsImdsv1ToImdsv2DashboardPerAccount",
        "amsImdsv1ToImdsv2DashboardSourceAccounts",
        "amsImdsv1ToImdsv2Remediation",
        "amsInfrastructureCICD",
        "amsModifyAlarmSNS",
        "amsOrphanedEBSVolumesCleanup",
        "amsPrismaCloud",
```

```
        "amsProwler",
        "amsPublicENIAudit",
        "amsQualysAgentManagement",
        "amsQuotaMonitor",
        "amsQuotaMonitorLogging",
        "amsRDPBastionPreWarm",
        "amsRDPBastionTools",
        "amsRdsCustomMonitoring",
        "amsRdsMSSQLMonitoring",
        "amsRDSSecretsRotation",
        "amsRFCSlackNotifications",
        "amsS3ReplicationCustomObjectKeys",
        "amsSnowflakeIntegration",
        "amsSpecificChangeTypeRFCNotification",
        "amsTagChecker",
        "amsUpdateR53onBastionRotation",
        "AWS_SSO_Running_on_AMS_MAD_Account"
      ]
    },
    "PatternParameters": {
      "description": "Add parameters (parameter name/value pairs) required for
deploying AMS Pattern.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Name": {
            "type": "string"
          },
          "Value": {
            "type": "string"
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Name",
            "Value"
          ]
        },
        "required": [
          "Name",
          "Value"
        ]
```

```
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "OrganizationalUnitIds": {
      "description": "Organizational Unit Ids in which the patterns will be deployed
 to. Use this for deploying a pattern as a StackSet stack in a multi-account landing
 zone (MALZ) Management account. For single-account landing zone (SALZ) application
 account, ignore this parameter.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "ou-[0-9a-z]{4,32}-[a-z0-9]{8,32}$"
      },
      "uniqueItems": true
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "PatternName",
      "PatternParameters",
      "OrganizationalUnitIds",
      "Priority"
    ]
  },
  "required": [
    "PatternName"
  ]
}
```

Schema for Change Type ct-2jvzjwunghrhy

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create (for Aurora)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create a RDS Aurora stack allowing either MultiAZ or Single Instance",
  "description": "Create an AWS Relational Database Service (RDS) Aurora stack using
 either multi-availability zone (MultiAZ) or a single instance.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
```

```
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-j24cifrdi0untnsn6",
      "type": "string",
      "enum": [
        "stm-j24cifrdi0untnsn6"
      ],
      "default": "stm-j24cifrdi0untnsn6"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AutoMinorVersionUpgrade": {
```

```
          "type": "string",
          "description": "True if the RDS instance should have automatic minor version
upgrade, false if it should not. Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "BackupRetentionPeriod": {
          "type": "integer",
          "description": "The number of days for which automatic database (DB)
snapshots are retained. Range is 1 - 35.",
          "default": 7,
          "minimum": 1,
          "maximum": 35
        },
        "ClusterName": {
          "type": "string",
          "description": "Optional identifier for the DB Cluster that is created with
your instance. If you do not provide one, a default identifier based on the instance
identifier is used. The cluster identifier is used in determining the cluster's
connection endpoint.",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$",
          "default": ""
        },
        "DBEngine": {
          "type": "string",
          "description": "The name of the engine for the Aurora database. For a MySQL
5.6 compatible database, use 'aurora', for a MySQL 5.7 compatible database, use
'aurora-mysql', for a PostgreSQL compatible database, use 'aurora-postgresql'. Not
every database engine is available for every AWS region. For a list of available
engines, use the DescribeDBEngineVersions AWS API action.",
          "enum": [
            "aurora",
            "aurora-mysql",
            "aurora-postgresql"
          ],
          "default": "aurora"
        },
        "DBName": {
          "type": "string",
          "description": "A name for the database. The meaning of this parameter
differs according to the database engine you use.",
```

```
          "pattern": "^[a-zA-Z0-9]{1,64}$",
          "maxLength": 64,
          "minLength": 1
        },
        "DBClusterParameterGroupName": {
          "description": "The name of an existing DB cluster parameter group. The
 parameter group must be compatible with the DBEngine and the EngineVersion.",
          "type": "string",
          "pattern": "^(?!.*--.*)(?!.*-$)[a-zA-Z][a-zA-Z0-9-.]{0,254}$"
        },
        "DBSubnetGroupName": {
          "type": "string",
          "description": "The name of an existing DB subnet group provisioned with the
 \"RDS database stack | Create DB subnet group\" change type.",
          "pattern": "^[a-zA-Z0-9._-]{1,255}$"
        },
        "EngineVersion": {
          "type": "string",
          "description": "The version number of the database engine to use. Not every
 database version is available for every AWS region.",
          "pattern": "^\\d.\\d.\\d{2}[a-z]$|^5.\\d.mysql_aurora.\\d.\\d{2}.\\d$|^8.\
\d.mysql_aurora.\\d.\\d{2}.\\d$|^(\\d{2}.\\d{0,2})$|^$",
          "default": ""
        },
        "InstanceType": {
          "type": "string",
          "description": "The instance type to use, this determines the compute and
 memory capacity for the DB instance. Not every instance type is available for every
 database engine.",
          "enum": [
            "db.serverless",
            "db.t2.small",
            "db.t2.medium",
            "db.t3.micro",
            "db.t3.small",
            "db.t3.medium",
            "db.t3.large",
            "db.t3.xlarge",
            "db.t3.2xlarge",
            "db.t4g.medium",
            "db.t4g.large",
            "db.r3.large",
            "db.r3.xlarge",
            "db.r3.2xlarge",
```

```
            "db.r3.4xlarge",
            "db.r3.8xlarge",
            "db.r4.large",
            "db.r4.xlarge",
            "db.r4.2xlarge",
            "db.r4.4xlarge",
            "db.r4.8xlarge",
            "db.r4.16xlarge",
            "db.r5.large",
            "db.r5.xlarge",
            "db.r5.2xlarge",
            "db.r5.4xlarge",
            "db.r5.8xlarge",
            "db.r5.12xlarge",
            "db.r5.16xlarge",
            "db.r5.24xlarge",
            "db.r6g.large",
            "db.r6g.xlarge",
            "db.r6g.2xlarge",
            "db.r6g.4xlarge",
            "db.r6g.8xlarge",
            "db.r6g.12xlarge",
            "db.r6g.16xlarge",
            "db.x2g.large",
            "db.x2g.xlarge",
            "db.x2g.2xlarge",
            "db.x2g.4xlarge",
            "db.x2g.8xlarge",
            "db.x2g.12xlarge",
            "db.x2g.16xlarge"
          ],
          "default": "db.r4.large"
        },
        "MasterUsername": {
          "type": "string",
          "description": "The name that you use with the configured MasterUserPassword
to log in to your DB instance. Must begin with a letter and contain from 1 to 16
alphanumeric characters.",
          "pattern": "^[a-zA-Z][a-zA-Z0-9]{1,15}$",
          "maxLength": 16,
          "minLength": 1
        },
        "MasterUserPassword": {
          "type": "string",
```

```
          "description": "The password that you use with the configured MasterUsername
 to log in to your DB instance. Must contain from 8 to 41 printable ASCII characters
 (excluding backslash, double quotes, and at sign).",
          "pattern": "^$|(?!@/\")[a-zA-Z0-9]{8,41}$",
          "maxLength": 41,
          "minLength": 8,
          "metadata": {
            "ams:sensitive": true
          }
        },
        "MultiAZ": {
          "type": "string",
          "description": "True to have a secondary replica of your DB instance created
 in another Availability Zone for failover support, false to not have a standby.
 Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "PerformanceInsights": {
          "type": "string",
          "description": "True to enable Performance Insights for the DB instance,
 false to not. Performance Insights is only available on engine type aurora and aurora-
postgresql.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "PerformanceInsightsKMSKey": {
          "type": "string",
          "description": "ARN of the KMS master key to use to encrypt Performance
 Insights data. Specify default to use the default RDS KMS Key.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
          "default": ""
        },
        "PerformanceInsightsRetentionPeriod": {
          "type": "string",
          "description": "The amount of time, in days, to retain Performance Insights
 data. Valid values are 7 or 731 (2 years).",
```

```
          "enum": [
            "7",
            "731"
          ],
          "default": "7"
        },
        "Port": {
          "type": "string",
          "description": "The port for the instance. Valid range is: 1150-65535.
 Specifying 0 assigns the default based on the selected DBEngine (aurora=3306, aurora-
mysql=3306, aurora-postgresql=5432).",
          "pattern": "^(0|11[5-8][0-9]|119[0-9]|1[2-9][0-9]{2}|[2-9][0-9]{3}|[1-5][0-9]
{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])$",
          "default": "0"
        },
        "PreferredBackupWindow": {
          "type": "string",
          "description": "The daily time range during which automated backups are
 created. Must be in the format hh:mm-hh:mm (24-hour format), in Universal Coordinated
 Time (UTC). Must not conflict with the PreferredMaintenanceWindow setting, and must be
 at least 30 minutes.",
          "pattern": "^[0-9]{2}:[0-9]{2}-[0-9]{2}:[0-9]{2}$",
          "default": "22:00-23:00"
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
          "description": "The weekly time range during which system maintenance
 can occur, in UTC. Must be in the format ddd:hh:mm-ddd:hh:mm (24-hour format), in
 Universal Coordinated Time (UTC) and must be at least 30 minutes. If you don't specify
 PreferredMaintenanceWindow, then Amazon RDS assigns a 30-minute maintenance window on
 a randomly selected day of the week.",
          "pattern": "^$|[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$",
          "default": ""
        },
        "ServerlessScalingMaxCapacity": {
          "description": "The maximum number of Aurora capacity units (ACUs) for a DB
 instance in an Aurora Serverless cluster. The largest value that you can use is 128.0.
 Only applies to db.serverless InstanceType.",
          "type": "number",
          "minimum": 1,
          "maximum": 128,
          "default": 1
        },
        "ServerlessScalingMinCapacity": {
```

```
          "description": "The minimum number of Aurora capacity units (ACUs) for a DB
 instance in an Aurora Serverless cluster. The smallest  value that you can use is 0.5.
 Only applies to db.serverless InstanceType.",
          "type": "number",
          "minimum": 0.5,
          "maximum": 128,
          "default": 0.5
        },
        "StorageEncryptionKey": {
          "type": "string",
          "description": "ARN of the KMS master key to use to encrypt the database.
 Specify default to use the default RDS KMS Key. Leave blank to not encrypt the
 database.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "DBEngine",
          "EngineVersion",
          "InstanceType",
          "MultiAZ",
          "DBName",
          "ClusterName",
          "DBClusterParameterGroupName",
          "DBSubnetGroupName",
          "MasterUsername",
          "MasterUserPassword",
          "Port",
          "StorageEncryptionKey",
          "AutoMinorVersionUpgrade",
          "PerformanceInsights",
          "PerformanceInsightsKMSKey",
          "PerformanceInsightsRetentionPeriod",
          "BackupRetentionPeriod",
          "PreferredBackupWindow",
          "PreferredMaintenanceWindow",
          "ServerlessScalingMaxCapacity",
          "ServerlessScalingMinCapacity"
        ]
      },
      "required": [
```

```
        "DBEngine",
        "EngineVersion",
        "DBName",
        "DBSubnetGroupName",
        "MasterUsername",
        "MasterUserPassword"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2lt0jeydeumpe

Classifications:

- Management | Advanced stack components | KMS key | Enable rotation

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable KMS CMK Auto Rotation",
```

```
  "description": "Enable automatic key rotation for an AWS Key Management Service (KMS)
 customer master key (CMK).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-EnableKMSKeyRotation.",
      "type": "string",
      "enum": [
        "AWSManagedServices-EnableKMSKeyRotation"
      ],
      "default": "AWSManagedServices-EnableKMSKeyRotation"
    },
    "Region": {
      "description": "The AWS Region in which the KMS Key is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "KeyId": {
          "description": "The ID of the KMS key to enable rotation for. This can be
 either the key ID or the key ARN.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/)?
[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "KeyId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "KeyId"
      ]
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2mf36chtp1ejh

Classifications:

- Management | Managed Firewall | Outbound (Palo Alto) | Remove URLs

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove Allow List URLs",
  "description": "Remove URLs from an allow list file for AMS managed Palo Alto
 firewall - Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be RemoveURLs.",
      "type": "string",
      "enum": [
        "RemoveURLs"
      ],
      "default": "RemoveURLs"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "URLs": {
```

```
          "description": "The URLs to remove from the allow list. URLs must end with a
 forward slash i.e '*.amazon.com/'.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((\\*|([a-zA-Z0-9][a-zA-Z0-9-_]{0,62}[a-zA-Z0-9]{0,1}))\\.)
{1,127}([a-zA-Z][a-zA-Z0-9\\-]{0,23}[a-zA-Z]\\/)$"
          },
          "minItems": 1,
          "maxItems": 50
        },
        "AllowListName": {
          "description": "The name of the allow list.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "URLs",
          "AllowListName"
        ]
      },
      "required": [
        "URLs",
        "AllowListName"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Parameters",
      "RequestType"
    ]
  },
  "required": [
    "Parameters",
    "RequestType"
  ]
}
```

Schema for Change Type ct-2murl5xzbxoxf

Classifications:

- Management | Directory Service | DNS | Add CNAME record


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add DNS CNAME Record",
  "description": "Create a new DNS CNAME record in AWS Managed Microsoft Active
 Directory (AD). CNAME records must always point to another domain name, never directly
 to an IP address. For multi-account landing zone (MALZ), use this change type in the
 shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-CreateDNSCnameRecord-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateDNSCnameRecord-Admin"
      ],
      "default": "AWSManagedServices-CreateDNSCnameRecord-Admin"
    },
    "Region": {
      "description": "The AWS Region where AWS managed Microsoft AD in Directory
 Service is located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RecordName": {
          "description": "Fully qualified domain name (FQDN) of the target host. For
 example, EC2WIN-testhost1.example.local or app-lb.elb.ap-southeast2.amazon.com.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9\\-\\.])+$"
          },
          "minItems": 1,
          "maxItems": 1
```

```
        },
        "RecordCname": {
          "description": "A meaningful name for the DNS CNAME record. For example,
 myapp1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-]{1,63}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "RecordName",
          "RecordCname"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RecordName",
        "RecordCname"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2ni31oyto1i5k

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create service-specific credentials


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Service-Specific Credentials",
  "description": "Generate a set of credentials consisting of a user name and password,
 to use to access the specified service.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateServiceSpecificCredentials.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateServiceSpecificCredentials"
      ],
      "default": "AWSManagedServices-CreateServiceSpecificCredentials"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Username": {
          "description": "The name of the IAM user to associate with the
 credentials.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[\\w+=,.@-]+"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
        "Service": {
          "description": "The name of the AWS service to associate with the
 credentials.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "CodeCommit"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Username",
          "Service"
        ]
      },
      "required": [
        "Username",
        "Service"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2nyeguspp2g1l

Classifications:

- Deployment | Patching | SSM patch baseline | Create (CentOS)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Baseline (CentOS)",
  "description": "Create an AWS Systems Manager (SSM) patch baseline to define which
 patches are approved for installation on your instances for CentOS. Specify existing
 instance \"Patch Group\" tag values for the patch baseline. The patch baseline is an
 SSM resource that you can manage with the SSM console.",
  "additionalProperties": false,
  "properties": {
    "ApprovalRules": {
      "description": "Create auto-approval rules to specify that certain types of
 operating system patches are approved automatically.",
      "items": {
        "additionalProperties": false,
        "properties": {
          "ApproveAfterDays": {
            "default": 7,
            "description": "The number of days to wait after a patch is released before
 approving patches automatically.",
            "maximum": 100,
            "minimum": 0,
            "type": "integer"
          },
          "Classification": {
            "description": "The Classification of the patches to be selected. Allowed
 values are \"All\", \"Bugfix\", \"Enhancement\", \"Newpackage\", \"Recommended\" and
 \"Security\".",
            "items": {
              "enum": [
                "All",
                "Bugfix",
                "Enhancement",
                "Newpackage",
                "Recommended",
                "Security"
              ],
```

```
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          },
          "Severity": {
            "description": "The severity of the patches to be selected. Allowed values
 are \"All\", \"Critical\", \"Important\", \"Low\", \"Moderate\" and \"None\".",
            "items": {
              "enum": [
                "All",
                "Critical",
                "Important",
                "Low",
                "Moderate",
                "None"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          }
        },
        "metadata": {
          "ui:order": [
            "Severity",
            "Classification",
            "ApproveAfterDays"
          ]
        },
        "required": [
          "ApproveAfterDays"
        ],
        "type": "object"
      },
      "maxItems": 10,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "ApprovedPatches": {
      "description": "The list of patches to approve explicitly.",
      "items": {
        "type": "string",
```

```
        "maxLength": 100,
        "minLength": 1
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Description": {
      "description": "A meaningful description for this patch baseline.",
      "maxLength": 500,
      "minLength": 1,
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch baseline.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "OperatingSystem": {
      "default": "CentOS",
      "description": "The operating system of instances to which this baseline is
applied.",
      "enum": [
        "CentOS"
      ],
      "type": "string"
    },
    "PatchGroupTagValues": {
      "description": "A list of the values of your \"Patch Group\" tags on the
instances you want patched; the values for up to twenty-five \"Patch Group\" tags can
be provided. Instances with those values are associated with this patch baseline.",
      "items": {
        "maxLength": 256,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 25,
      "minItems": 1,
      "type": "array",
      "uniqueItems": true
    },
```

```
    "RejectedPatches": {
      "description": "The list of patches to reject explicitly.",
      "items": {
        "maxLength": 100,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the SSM patch
 baseline resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
```

```
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "OperatingSystem",
      "Name",
      "Description",
      "PatchGroupTagValues",
      "ApprovalRules",
      "ApprovedPatches",
      "RejectedPatches",
      "Tags"
    ]
  },
  "required": [
    "Name",
    "PatchGroupTagValues",
    "OperatingSystem"
  ],
  "type": "object"
}
```

Schema for Change Type ct-2oxl37nphsrjz

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create source endpoint (S3)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS source endpoint for S3",
  "description": "Use to create a Database Migration Service (DMS) source endpoint for
 S3.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
```

```
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
```

```
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-pud4ghhkp7395n9bc.",
      "type": "string",
      "enum": [
        "stm-pud4ghhkp7395n9bc"
      ],
      "default": "stm-pud4ghhkp7395n9bc"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "EndpointIdentifier": {
          "type": "string",
          "description": "A meaningful identifier for the source database endpoint.
Must be unique for all endpoints owned by your AWS account in the current region. Must
begin with a letter, must contain only ASCII letters, digits and hyphens and must not
end with a hyphen or contain two consecutive hyphens.",
          "pattern": "^$|(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$",
          "default": ""
        },
        "EngineName": {
          "type": "string",
          "description": "Must be s3.",
          "enum": [
            "s3"
          ]
        },
        "ExtraConnectionAttributes": {
          "type": "string",
```

```
        "description": "Additional attributes associated with the connection. See AWS
documentation for more information on the supported extra connection attributes for
S3.",
        "default": ""
      },
      "S3BucketFolder": {
        "type": "string",
        "description": "The folder name in the S3 bucket. This is the Amazon S3
bucket path where the CSV files can be found."
      },
      "S3BucketName": {
        "type": "string",
        "description": "The name of the Amazon S3 bucket."
      },
      "S3CompressionType": {
        "type": "string",
        "description": "Type of compression to use.",
        "enum": [
          "GZIP",
          "NONE"
        ],
        "default": "NONE"
      },
      "S3CsvDelimiter": {
        "type": "string",
        "description": "The delimiter used to separate columns in the source files.
The default is a comma."
      },
      "S3CsvRowDelimiter": {
        "type": "string",
        "description": "The delimiter used to separate rows in the source files. The
default is a carriage return (\\n)"
      },
      "S3ExternalTableDefinition": {
        "type": "string",
        "description": "The definition of the external table. A JSON document
describing the structure of the tables and columns in the CSV files."
      },
      "S3ServiceAccessRoleArn": {
        "type": "string",
        "description": "The Amazon Resource Name (ARN) of the service access IAM
role.",
        "pattern": "^$|^arn:aws:iam::[0-9]{12}:role/[\\w-]+$"
      }
```

```
      },
      "metadata": {
        "ui:order": [
          "EndpointIdentifier",
          "EngineName",
          "ExtraConnectionAttributes",
          "S3BucketName",
          "S3BucketFolder",
          "S3CompressionType",
          "S3CsvDelimiter",
          "S3CsvRowDelimiter",
          "S3ExternalTableDefinition",
          "S3ServiceAccessRoleArn"
        ]
      },
      "required": [
        "EngineName",
        "S3BucketName",
        "S3ExternalTableDefinition",
        "S3ServiceAccessRoleArn"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
```

```
}
```

Schema for Change Type ct-2p93tyd5angmi

Classifications:

- Deployment | Managed landing zone | Management account | Create Accelerate account

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Accelerate Account",
  "description": "Create an Accelerate account in your AMS-managed landing zone.
 Accelerate provides patching, backup, monitoring and reports, but no requests for
 change.",
  "type": "object",
  "properties": {
    "AccountName": {
      "description": "A name for the new Accelerate account. Max length 50 characters.
 The underscore (_) is not allowed.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9]{1}[a-zA-Z0-9.-]{0,49}$"
    },
    "AccountEmail": {
      "description": "The email address for the new Accelerate account. The email must
 be unique per account.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
    },
    "SupportLevel": {
      "description": "The account's AMS support level, Premium or Plus.",
      "type": "string",
      "enum": [
        "plus",
        "premium"
      ]
    },
    "AccelerateOUName": {
      "description": "The name of an existing organizational unit (OU) for this
 Accelerate account, default is accelerate. To use a child OU of an existing OU, the
 format is <Accelerate OU name>:<child OU name>.",
      "type": "string",
      "default": "accelerate"
```

```
    },
    "Regions": {
      "description": "Select the AWS Region or Regions that you want AMS Accelerate to
 manage. The primary Region, the Region of your MALZ environment, must be included.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^(ap-northeast-1|ap-northeast-2|ap-south-1|ap-southeast-1|ap-
southeast-2|ca-central-1|eu-central-1|eu-north-1|eu-west-1|eu-west-2|eu-west-3|sa-
east-1|us-east-1|us-east-2|us-west-1|us-west-2)$"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "EnablePatch": {
      "description": "True to enable patch add-on, false to not. For an AWS account
 with the patch add-on, AMS monitors, reports, and installs vendor updates to EC2
 instances for supported operating systems during your chosen maintenance windows.
 Please consult your CSDM about the charges for the Patch add-on.",
      "type": "boolean",
      "default": false
    }
  },
  "metadata": {
    "ui:order": [
      "AccountName",
      "AccountEmail",
      "AccelerateOUName",
      "Regions",
      "SupportLevel",
      "EnablePatch"
    ]
  },
  "additionalProperties": false,
  "required": [
    "AccountName",
    "AccountEmail",
    "AccelerateOUName",
    "Regions",
    "SupportLevel",
    "EnablePatch"
  ]
}
```

Schema for Change Type ct-2paw0y79kvr3l

Classifications:

- Management | Managed landing zone | Application account | Delete VPC

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Application Account VPC",
  "description": "Delete the virtual private cloud (VPC) in a managed landing zone
 application account.",
  "type": "object",
  "properties": {
    "VPCId": {
      "description": "The ID of the VPC to be deleted.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    }
  },
  "metadata": {
    "ui:order": [
      "VPCId"
    ]
  },
  "additionalProperties": false,
  "required": [
    "VPCId"
  ]
}
```

Schema for Change Type ct-2pbqoffhclpek

Classifications:

- Management | Advanced stack components | Route 53 Resolver | Associate VPC with resolver rule

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Associate VPC With Resolver Rule",
```

```
 "description": "Associate a VPC with a Route 53 resolver rule, this causes the
resolver to forward all DNS queries for the domain name specified in the rule, and
that originate in the VPC, to the IP addresses specified in the rule.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-AssociateVPCWithResolverRule.",
     "type": "string",
     "enum": [
       "AWSManagedServices-AssociateVPCWithResolverRule"
     ],
     "default": "AWSManagedServices-AssociateVPCWithResolverRule"
   },
   "Region": {
     "description": "The AWS Region in which the Route 53 Resolver Rule is located, in
the form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "Name": {
         "description": "A name for the association that you're creating between the
resolver rule and a VPC.",
         "type": "string",
         "pattern": "^$|^(?!.*(AWSManagedServices-|AMS-|ams-))[A-Za-z0-9-_' ']+$",
         "default": ""
       },
       "ResolverRuleId": {
         "description": "The ID of the resolver rule that you want to associate with
the VPC.",
         "type": "string",
         "pattern": "^(rslvr-rr-)[a-zA-Z0-9]{1,64}$"
       },
       "VPCId": {
         "description": "The ID of the VPC that you want to associate the resolver
rule with.",
         "type": "string",
         "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
       }
     },
     "metadata": {
       "ui:order": [
```

```
          "Name",
          "ResolverRuleId",
          "VPCId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResolverRuleId",
        "VPCId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2pfarpvczsstr

Classifications:

- Management | Advanced stack components | Route 53 Resolver | Disassociate resolver rules from VPC

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disassociate resolver rules from VPC",
  "description": "Removes the associations between specified resolver rules (upto 20)
 and a specified VPC.",
  "type": "object",
  "properties": {
```

```
    "DocumentName": {
      "description": "Must be AWSManagedServices-DisassociateVPCResolverRules.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DisassociateVPCResolverRules"
      ],
      "default": "AWSManagedServices-DisassociateVPCResolverRules"
    },
    "Region": {
      "description": "The AWS Region in which the Route 53 Resolver Rule is located, in
the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ResolverRuleIds": {
          "description": "A list of resolver rule IDs that you want to disassociate
from the VPC.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(rslvr-rr-)[a-zA-Z0-9]{1,64}$"
          },
          "minItems": 1,
          "maxItems": 20
        },
        "VPCId": {
          "description": "The ID of the VPC where Route53 resolver rules are
associated.",
          "type": "string",
          "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
        }
      },
      "metadata": {
        "ui:order": [
          "ResolverRuleIds",
          "VPCId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResolverRuleIds",
```

```
        "VPCId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2pkdckieh62ps

Classifications:

- Management | AMS Resource Scheduler | Period | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Resource Scheduler Period",
  "description": "Update an existing period used in AMS Resource Scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddOrUpdatePeriod.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddOrUpdatePeriod"
      ],
      "default": "AWSManagedServices-AddOrUpdatePeriod"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
 solution is, in the form us-east-1.",
```

```
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Action": {
          "description": "Specify the value: update. This explicitly requests that the
Resource Scheduler period be updated. The option cannot be left blank; it must be
update.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "update"
            ],
            "default": "update"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Name": {
          "description": "The name of the period to update.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/])^[A-Za-z0-9-_, +=.:#/]{1,64}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Description": {
          "description": "A meaningful description for the period.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,1000}$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "BeginTime": {
          "description": "The time, in HH:MM format, a resource starts under this
period.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((?:[01]\\d|2[0-3]):[0-5]\\d)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "EndTime": {
          "description": "The time, in HH:MM format, a resource stops under this
period.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((?:[01]\\d|2[0-3]):[0-5]\\d)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Months": {
          "description": "Enter a comma-delimited list of months (e.g. jan, feb), a
hyphenated range of months (e.g. jan-dec), or every n-th month (e.g. jan/3 for every
3rd month starting from jan) during which the resource runs. Abbreviated month names
(e.g. jan, feb, march) and numbers (1, 2, 12) are supported.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,-/]*)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "MonthDays": {
          "description": "Enter a comma-delimited list of days of the month (e.g. 1,
5, 15), a hyphenated range of days (e.g. 1-15), every n-th day of the month (e.g 1/7
for every 7th day starting on the 1st) or every n-th day day of the month in a range
( e.g. 1-15/2 for every other day from 1st to the 15th), the last day of the month
(specify L), or the nearest weekday to a specific date (specify W e.g. 15W) during
which the resource runs.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,-/]*)$|^$"
          },
```

```
          "maxItems": 1,
          "minItems": 1
        },
        "WeekDays": {
          "description": "Enter a comma-delimited list of days of the week (e.g.
Mon, Wed, Fri), a range of days of the week (e.g. Mon-Thu), or n-th occurrence
of a weekday in the month (e.g Mon#1 or 0#1 for first Monday of the month) during
which the resource runs. Enter a day and L ro run a resource on the last occurrence
of that weekday in the month (e.g. friL or 4L to run on the last Friday of the
month). Abbreviated week day names (e.g. Sun, Mon, Thu), and numbers (0, 1, 3), are
supported.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_,/]$)^([a-zA-Z0-9,#-/]*)$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Action",
          "Name",
          "Description",
          "BeginTime",
          "EndTime",
          "Months",
          "MonthDays",
          "WeekDays"
        ]
      },
      "required": [
        "Action",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
```

```
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2ptn20pq7ur3x

Classifications:

- Management | AMS Resource Scheduler | Schedule | Describe

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Describe Resource Scheduler Schedules",
  "description": "Describe (generate a detailed list) of existing schedules used in AMS
 Resource Scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DescribeScheduleOrPeriods.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DescribeScheduleOrPeriods"
      ],
      "default": "AWSManagedServices-DescribeScheduleOrPeriods"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
 solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ConfigurationType": {
```

```
          "description": "Specify the value: schedules. This explicitly requests that
 the Resource Scheduler existing schedules be described. The option cannot be left
 blank; it must be schedules.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "schedules"
            ],
            "default": "schedules"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ConfigurationType"
        ]
      },
      "required": [
        "ConfigurationType"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2pxyajek47am2

Classifications:

- Management | Managed landing zone | Networking account | Disable TGW propagation


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disable TGW Propagation",
  "description": "Disable the Transit Gateway (TGW) attachment from propagating routes
 to the TGW route table. For multi-account landing zone (MALZ), use this change type in
 the Network account only.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DisableTGWRouteTablePropagation.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DisableTGWRouteTablePropagation"
      ],
      "default": "AWSManagedServices-DisableTGWRouteTablePropagation"
    },
    "Region": {
      "description": "The AWS Region where the TGW attachment and TGW route table are
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "TransitGatewayAttachmentId": {
          "description": "The TGW attachment ID, in the form tgw-
attach-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-attach-[a-z0-9]{17}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
```

```
        "TransitGatewayRouteTableId": {
          "description": "The TGW route table ID, in the form tgw-
rtb-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-rtb-[a-z0-9]{17}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "TransitGatewayAttachmentId",
          "TransitGatewayRouteTableId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayAttachmentId",
        "TransitGatewayRouteTableId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2q5azjd8p1ag5

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication subnet group


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create a DMS replication subnet group",
  "description": "Use to create a Database Migration Service (DMS) replication subnet
 group. Resource creation will fail if the dms-vpc-role IAM role doesn't already
 exist.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true,
      "items": {
        "type": "object",
```

```
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      }
    },
    "StackTemplateId": {
      "description": "Must be stm-j637f96ls1h4oy5fj",
      "type": "string",
      "enum": [
        "stm-j637f96ls1h4oy5fj"
      ]
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
        "Identifier": {
          "type": "string",
          "description": "The identifier for the replication subnet group. Given a
unique ID if none is provided.",
          "pattern": "[0-9a-zA-Z\\-]{0,255}"
        },
        "Description": {
          "type": "string",
          "description": "The description for the replication subnet group.",
          "pattern": "[^\\n]+"
        },
        "SubnetIds": {
          "type": "array",
          "description": "Two or more subnet IDs for the replication subnet group, in
the form subnet-0123abcd or subnet-01234567890abcdef.",
          "items": {
            "type": "string"
          }
        }
      },
      "metadata": {
        "ui:order": [
          "SubnetIds",
          "Identifier",
          "Description"
        ]
      },
      "required": [
        "Description",
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
```

```
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2qhl8j1pjnbgn

Classifications:

- Deployment | Directory Service | DNS | Create group managed service account

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Group Managed Service Account",
  "description": "Create a new Active Directory (AD) Group Managed Service Account
 (gMSA). For multi-account landing zone (MALZ), use this change type in the shared
 services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateADGroupManagedServiceAccount-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateADGroupManagedServiceAccount-Admin"
      ],
      "default": "AWSManagedServices-CreateADGroupManagedServiceAccount-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
```

```
      "type": "object",
      "properties": {
        "AccountName": {
          "description": "A meaningful name for your service account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ComputerName": {
          "description": "The name of the computer object that will be added as a
member to the AD group provided in the parameter PrincipalAllowedToRetrievePassword.
If you are using this parameter, then you must also provide the
'PrincipalAllowedToRetrievePassword' parameter.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DNSHostName": {
          "description": "The fully qualified DNS host name of the AD Group Managed
Service Account (gMSA).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+[.]?$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ManagedPasswordIntervalInDays": {
          "description": "The number of days before a password change is required.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\d+$",
            "default": "30"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        },
        "PrincipalAllowedToRetrievePassword": {
          "description": "AD Group or principal that can retrieve the gMSA password
from the Domain Controller.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_\\ ]*[\\$]?$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KerberosEncryptionType": {
          "description": "The Kerberos encryption types the service account
supports. If this parameter is empty, the encryption supported will be set to
RC4,AES128,AES256",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^(RC4|AES128|AES256|None)(,(RC4|AES128|AES256|None))*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "AccountName",
          "ManagedPasswordIntervalInDays",
          "PrincipalAllowedToRetrievePassword",
          "ComputerName",
          "DNSHostName",
          "KerberosEncryptionType"
        ]
      },
      "additionalProperties": false,
      "required": [
        "AccountName"
      ]
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2qjqju7h67s7w

Classifications:

- Management | Monitoring and notification | GuardDuty threat intel set | Delete (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete GuardDuty ThreatIntelSet",
  "description": "Use to delete an Amazon GuardDuty ThreatIntelSet instance which is a
 list of known malicious IP addresses.",
  "type": "object",
  "properties": {
    "DetectorId": {
      "description": "The detector ID that specifies the GuardDuty service whose
 ThreatIntelSet you want to delete. Leave this blank to use the only detector in the
 selected region (this will not succeed if there is more than one detector in the
 selected region).",
      "pattern": "^[a-fA-F0-9]{32}$|^$",
      "type": "string"
    },
    "Region": {
      "description": "Region to use in the form of us-east-1.",
      "type": "string",
      "minLength": 1
    },
    "ThreatIntelSetId": {
```

```
      "description": "The unique ID that specifies the ThreatIntelSet that you want to
 delete.",
      "type": "string",
      "minLength": 1
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "ThreatIntelSetId",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "ThreatIntelSetId"
  ]
}
```

Schema for Change Type ct-2qldv4h9osmau

Classifications:

- Deployment | Advanced stack components | Network Load Balancer | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Network Load Balancer",
  "description": "Use to create a Network Load Balancer.",
  "type": "object",
```

```
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "minItems": 0,
     "maxItems": 40,
     "uniqueItems": true,
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
           "minLength": 1,
           "maxLength": 127
         }
       },
       "additionalProperties": false,
       "metadata": {
```

```
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      }
    },
    "StackTemplateId": {
      "description": "Must be stm-[a-z]{17}",
      "type": "string",
      "enum": [
        "stm-l70qr9itukvqssg8d"
      ],
      "default": "stm-l70qr9itukvqssg8d"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "HealthCheckHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check successes required to
declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10",
          "default": "3"
        },
        "HealthCheckIntervalSeconds": {
          "type": "string",
          "description": "The approximate interval, in seconds, between health
checks.",
          "enum": [
            "10",
```

```
            "30"
          ],
          "default": "30"
        },
        "HealthCheckTargetPath": {
          "type": "string",
          "description": "The ping path destination on the application hosts
where the load balancer sends health check requests. This is only applicable if
HealthCheckTargetProtocol = HTTP or HTTPS.",
          "default": "/"
        },
        "HealthCheckTargetPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health checks
on targets. The default is traffic-port, which indicates the port on which each target
receives traffic from the load balancer.",
          "pattern": "([0-9]{1,5})?",
          "default": ""
        },
        "HealthCheckTargetProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS",
            "TCP"
          ],
          "default": "TCP"
        },
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
instance.",
          "pattern": "[0-9]{1,5}",
          "default": "80"
        },
        "LoadBalancerName": {
          "type": "string",
          "description": "A friendly name for the load balancer."
        },
        "LoadBalancerPort": {
          "type": "string",
```

```
        "description": "The port number for the load balancer to use when routing
external incoming traffic.",
        "pattern": "[0-9]{1,5}",
        "default": "80"
      },
      "Public": {
        "type": "string",
        "description": "True if the load balancer endpoint is public, false if it
is not. Default is false. Set to true if you choose a public subnet for the load
balancer.",
        "enum": [
          "true",
          "false"
        ],
        "default": "false"
      },
      "CrossZoneEnabled": {
        "type": "string",
        "description": "True if cross-zone load balancing is enabled. False if it is
not.",
        "enum": [
          "true",
          "false"
        ],
        "default": "false"
      },
      "SubnetIds": {
        "type": "array",
        "description": "One or more subnet IDs for the load balancer, in the form
subnet-0123abcd or subnet-01234567890abcdef.",
        "items": {
          "type": "string"
        }
      },
      "ProxyProtocolV2": {
        "type": "string",
        "description": "True if proxy protocol version 2 is enabled. False if it is
not.",
        "enum": [
          "true",
          "false"
        ],
        "default": "false"
      },
```

```
      "DeregistrationDelayTimeoutSeconds": {
        "type": "string",
        "description": "The amount of time, in seconds, for Elastic Load Balancing to
wait before changing the state of a deregistering target from draining to unused.",
        "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})",
        "default": "300"
      },
      "TargetType": {
        "type": "string",
        "description": "The registration type of the targets in this target group.",
        "enum": [
          "instance",
          "ip"
        ],
        "default": "instance"
      },
      "Target1ID": {
        "type": "string",
        "description": "ID of the EC2 instance to register a target, in the form
i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
ip. Leave blank if you don't need to register a target.",
        "default": ""
      },
      "Target1Port": {
        "type": "string",
        "description": "The port number on which the target is listening for
traffic.",
        "default": ""
      },
      "Target1AvailabilityZone": {
        "type": "string",
        "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target1ID is
outside the VPC. Leave blank if TargetType = instance.",
        "default": ""
      },
      "Target2ID": {
        "type": "string",
        "description": "ID of the EC2 instance to register a target, in the form
i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
ip. Leave blank if you don't need to register a target.",
        "default": ""
```

```
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic.",
          "default": ""
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target2ID is
outside the VPC. Leave blank if TargetType = instance.",
          "default": ""
        },
        "Target3ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
ip. Leave blank if you don't need to register a target.",
          "default": ""
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic.",
          "default": ""
        },
        "Target3AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target3ID is
outside the VPC. Leave blank if TargetType = instance.",
          "default": ""
        },
        "Target4ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
ip. Leave blank if you don't need to register a target.",
          "default": ""
```

```
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
traffic.",
          "default": ""
        },
        "Target4AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. Use an Availability
Zone name if the target receives traffic from the load balancer nodes in the specified
Availability Zone. Use all if the traffic is received from all enabled Availability
Zones for the load balancer and the TargetType = ip and the IP address in Target4ID is
outside the VPC. Leave blank if TargetType = instance.",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "LoadBalancerName",
          "SubnetIds",
          "Public",
          "LoadBalancerPort",
          "InstancePort",
          "ProxyProtocolV2",
          "DeregistrationDelayTimeoutSeconds",
          "CrossZoneEnabled",
          "HealthCheckTargetPath",
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckIntervalSeconds",
          "TargetType",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
```

```
          "Target4AvailabilityZone"
        ]
      },
      "required": [
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2r2bffv9u6q4m

Classifications:

- Management | Advanced stack components | RDS database stack | Stop DB instance

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop RDS DB Instance",
  "description": "Stop an Amazon Relational Database Service (RDS) database (DB)
 instance. After seven days, the DB instance is automatically re-started. Supported
```

```
 engines are: MariaDB, Microsoft SQL Server, MySQL, Oracle, PostgreSQL. This change
 type doesn't apply to Aurora MySQL and Aurora PostgreSQL.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StopRDSInstance.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StopRDSInstance"
      ],
      "default": "AWSManagedServices-StopRDSInstance"
    },
    "Region": {
      "description": "The AWS Region in which the RDS DB is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "RDS DB instance identifier.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?=[a-zA-Z0-9-]{1,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId"
      ]
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2r9xvd3sdsic0

Classifications:

- Management | Managed account | Automated IAM provisioning with read-write permissions | Update custom deny list (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update custom deny list for Automated IAM Provisioning",
  "description": "Update the list of customer-defined denied actions for Automated
 IAM Provisioning. Make sure to provide the complete list of deny actions, including
 previously provisioned actions. The provided list replaces the previous list.",
  "type": "object",
  "properties": {
    "CustomerCustomDenyActionsList1": {
      "description": "A comma-separated list of actions to update the custom deny list.
 For example 'ec2:RunInstances, s3:Get*'. These actions will be denied in IAM policies
 created or updated by Automated IAM provisioning.",
      "type": "string",
      "pattern": "^[a-z0-9-]+:[A-Za-z0-9*-]+(?:,[a-z0-9-]+:[A-Za-z0-9*-]+)*$",
      "maxLength": 4096
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
```

```
      "default": "High",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "CustomerCustomDenyActionsList1",
      "Priority"
    ]
  },
  "required": [
    "CustomerCustomDenyActionsList1"
  ]
}
```

Schema for Change Type ct-2rfzmkm6ugigh

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Delete account alias

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete AWS Account Alias",
  "description": "Delete an existing AWS account alias. Note that if you delete the
 account alias, any URL containing the account alias stops working.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteAccountAlias.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteAccountAlias"
      ],
      "default": "AWSManagedServices-DeleteAccountAlias"
```

```
    },
    "Region": {
      "description": "The AWS Region where the account is, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AWSAccountAlias": {
          "description": "The alias name of the AWS account to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?=[a-zA-Z0-9-]{3,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AWSAccountAlias"
        ]
      },
      "required": [
        "AWSAccountAlias"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
```

```
}
```

Schema for Change Type ct-2rnjx5yd6jgpt

Classifications:

- Management | Monitoring and notification | GuardDuty threat intel set | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update GuardDuty ThreatIntelSet",
  "description": "Use to update an Amazon GuardDuty ThreatIntelSet instance which
 is a list of trusted IP addresses that have been whitelisted for highly secure
 communication with your AWS environment.",
  "type": "object",
  "properties": {
    "Activate": {
      "description": "Specified whether the ThreatIntelSet is active or not.",
      "type": "boolean",
      "default": true
    },
    "DetectorId": {
      "description": "The detector ID that specifies the GuardDuty service to which
 you want to update an ThreatIntelSet. Leave this blank to use the only detector in
 the selected region (this will not succeed if there is more than one detector in the
 selected region).",
      "pattern": "^[a-fA-F0-9]{32}$|^$",
      "type": "string"
    },
    "ThreatIntelSet": {
      "description": "The URI of the file that contains the ThreatIntelSet.",
      "minLength": 1,
      "type": "string"
    },
    "ThreatIntelSetId": {
      "description": "The unique ID that specifies the ThreatIntelSet that you want to
 update.",
      "type": "string",
      "minLength": 1
    },
    "Name": {
```

```
      "description": "The friendly name to identify the ThreatIntelSet. This name is
 displayed in all findings that are triggered by activity that involves IP addresses
 included in this ThreatIntelSet.",
      "minLength": 1,
      "type": "string"
    },
    "Region": {
      "description": "The region containing the GuardDuty detector to use; in the form
 of us-east-1.",
      "minLength": 1,
      "type": "string"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "ThreatIntelSetId",
      "Name",
      "ThreatIntelSet",
      "Activate",
      "DetectorId",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "ThreatIntelSetId",
    "Region"
  ]
}
```

Schema for Change Type ct-2svg4k2fqi4ak

Classifications:

- Deployment | Advanced stack components | KMS alias | Create


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create KMS Alias",
  "description": "Create an alias for an AWS Key Management Service (KMS) customer
 master key (CMK).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateKMSAlias.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateKMSAlias"
      ],
      "default": "AWSManagedServices-CreateKMSAlias"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AliasName": {
          "description": "Alias name. The value must not start with aws/. Don't specify
 the prefix alias/, it will be added during the execution.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!alias/)(?!(mc|MC|ams|AMS|aws|AWSManagedServices))[a-zA-
Z0-9/_-]{1,250}"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
        "TargetKeyId": {
          "description": "The ID of the KMS key to create the alias for.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/)?[a-f0-9]{8}-[a-f0-9]
{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "AliasName",
          "TargetKeyId"
        ]
      },
      "required": [
        "AliasName",
        "TargetKeyId"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2syhk4sr7cvyw

Classifications:

- Management | Advanced stack components | RDS database stack | Update deletion protection


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Deletion Protection setting for RDS instance or cluster",
  "description": "Update the DeletionProtection setting for the specified RDS
 instance or cluster. The RDS instance or cluster can be standalone or belong to a
 CloudFormation stack; in the latter case, the change might cause stack drift. To avoid
 causing stack drift, use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the RDS was
 provisioned through CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateRDSDeletionProtection.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateRDSDeletionProtection"
      ],
      "default": "AWSManagedServices-UpdateRDSDeletionProtection"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBIdentifierArn": {
          "description": "The Amazon Resource Name (ARN) of the RDS instance or
 cluster.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{12}:(db|cluster):[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
        },
        "DeletionProtection": {
```

```
          "description": "True to enable DeletionProtection, false to disable
 DeletionProtection. Use this to change the current DeletionProtection status.",
          "type": "boolean"
        }
      },
      "metadata": {
        "ui:order": [
          "DBIdentifierArn",
          "DeletionProtection"
        ]
      },
      "required": [
        "DBIdentifierArn",
        "DeletionProtection"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2taqdgegqthjr

Classifications:

- Deployment | Patching | SSM patch baseline | Create (Amazon Linux)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Baseline (Amazon Linux)",
```

```
 "description": "Create an AWS Systems Manager (SSM) patch baseline to define which
patches are approved for installation on your instances for Amazon Linux OS. Specify
existing instance \"Patch Group\" tag values for the patch baseline. The patch
baseline is an SSM resource that you can manage with the SSM console.",
 "additionalProperties": false,
 "properties": {
   "ApprovalRules": {
     "description": "Create auto-approval rules to specify that certain types of
operating system patches are approved automatically.",
     "items": {
       "additionalProperties": false,
       "properties": {
         "ApproveAfterDays": {
           "default": 7,
           "description": "The number of days to wait after a patch is released before
approving patches automatically.",
           "maximum": 100,
           "minimum": 0,
           "type": "integer"
         },
         "Classification": {
           "description": "The Classification of the patches to be selected. Allowed
values are \"All\", \"Bugfix\", \"Enhancement\", \"Newpackage\", \"Recommended\" and
\"Security\".",
           "items": {
             "enum": [
               "All",
               "Bugfix",
               "Enhancement",
               "Newpackage",
               "Recommended",
               "Security"
             ],
             "type": "string"
           },
           "type": "array",
           "uniqueItems": true
         },
         "Severity": {
           "description": "The severity of the patches to be selected. Allowed values
are \"All\", \"Critical\", \"Important\", \"Low\" and \"Medium\".",
           "items": {
             "enum": [
               "All",
```

```
              "Critical",
              "Important",
              "Low",
              "Medium"
            ],
            "type": "string"
          },
          "type": "array",
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "Severity",
          "Classification",
          "ApproveAfterDays"
        ]
      },
      "required": [
        "ApproveAfterDays"
      ],
      "type": "object"
    },
    "maxItems": 10,
    "minItems": 0,
    "type": "array",
    "uniqueItems": true
  },
  "ApprovedPatches": {
    "description": "The list of patches to approve explicitly.",
    "items": {
      "type": "string",
      "maxLength": 100,
      "minLength": 1
    },
    "maxItems": 50,
    "minItems": 0,
    "type": "array",
    "uniqueItems": true
  },
  "Description": {
    "description": "A meaningful description for this patch baseline.",
    "maxLength": 500,
    "minLength": 1,
```

```
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch baseline.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "OperatingSystem": {
      "default": "Amazon Linux",
      "description": "The operating system of instances to which this baseline is
applied.",
      "enum": [
        "Amazon Linux"
      ],
      "type": "string"
    },
    "PatchGroupTagValues": {
      "description": "A list of the values of your \"Patch Group\" tags on the
instances you want patched; the values for up to twenty-five \"Patch Group\" tags can
be provided. Instances with those values are associated with this patch baseline.",
      "items": {
        "maxLength": 256,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 25,
      "minItems": 1,
      "type": "array",
      "uniqueItems": true
    },
    "RejectedPatches": {
      "description": "The list of patches to reject explicitly.",
      "items": {
        "maxLength": 100,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
```

```
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the SSM patch
 baseline resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "OperatingSystem",
      "Name",
      "Description",
      "PatchGroupTagValues",
      "ApprovalRules",
      "ApprovedPatches",
      "RejectedPatches",
```

```
      "Tags"
    ]
  },
  "required": [
    "Name",
    "PatchGroupTagValues",
    "OperatingSystem"
  ],
  "type": "object"
}
```

Schema for Change Type ct-2tqi3kjcusen4

Classifications:

- Management | Managed account | DNS | Migrate to Route 53

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Migrate AWS Managed Microsoft AD to Route 53 DNS resolver for SALZ
 accounts",
  "description": "Change the DNS resolution in your Amazon VPC by enabling Route 53
 as the default DNS resolver for your SALZ account. This transition from Microsoft
 AD to Route 53 Resolver involves redirecting DNS traffic within your VPC through
 strategically implemented Route 53 Resolver Endpoints and Conditional Forwarders.
 These forwarders act as rules to intelligently route DNS queries, ensuring seamless
 resolution for various destinations. It's essential to plan the migration during
 a scheduled maintenance window to minimize potential disruptions caused by DNS
 changes.",
  "type": "object",
  "properties": {
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
```

```
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Priority"
    ]
  },
  "required": [

  ]
}
```

Schema for Change Type ct-2tylseo8rxfsc

Classifications:

- Deployment | Advanced stack components | Auto Scaling group | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Auto Scaling group",
  "description": "Use to create an Auto Scaling group, the launch configuration to
 use to create new instances when needed, and CloudWatch metrics and alarms for the
 group.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "The ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackTemplateId": {
      "description": "Must be stm-suw38u40000000000.",
      "type": "string",
      "enum": [
        "stm-suw38u40000000000"
```

```
      ]
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to seven tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 7,
      "uniqueItems": true
    },
```

```
   "TimeoutInMinutes": {
     "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
     "type": "integer",
     "minimum": 0,
     "maximum": 360
   },
   "Parameters": {
     "description": "Specifications for the stack.",
     "type": "object",
     "properties": {
       "ASGAmiId": {
         "description": "The AMI for the Auto Scaling group to use when creating new
instances, in the form ami-0123abcd or ami-01234567890abcdef.",
         "type": "string",
         "pattern": "^ami-[a-z0-9]{8}$|^ami-[a-z0-9]{17}$"
       },
       "ASGCooldown": {
         "description": "The number of seconds after a scaling activity is complete
before any further scaling activities can start.",
         "type": "integer",
         "minimum": 120,
         "maximum": 600,
         "default": 300
       },
       "ASGDesiredCapacity": {
         "description": "The number of EC2 instances you want running in the group.
This number must be greater than or equal to the ASGMinInstances setting and less than
or equal to the ASGMaxInstances setting.",
         "type": "integer",
         "minimum": 1,
         "maximum": 1000,
         "default": 1
       },
       "ASGEBSOptimized": {
         "description": "True to create EBS-optimized instances, false to not.
EBS-optimization provides dedicated throughput to Amazon EBS and optimal EBS I/O
performance.",
         "type": "boolean",
         "default": false
       },
       "ASGHealthCheckGracePeriod": {
```

```
        "description": "The amount of time, in seconds, that Auto Scaling waits
before checking the health status of an EC2 instance that has come into service.
During this time, any health check failures for the instance are ignored.",
        "type": "integer",
        "minimum": 600,
        "maximum": 1800,
        "default": 1800
      },
      "ASGHealthCheckType": {
        "description": "The service to use for the health checks. The ELB
Health Check Type includes EC2 instance and system status checks. Only choose
ELB as the ASGHealthCheckType if the ASG is being fronted by Load Balancers. If
ASGHealthCheckType = ELB, ensure that your ASGHealthCheckGracePeriod value is long
enough so that your instances are not terminated due to load-balancer health checks
failing, before your application has been deployed.",
        "default": "EC2",
        "type": "string",
        "enum": [
          "EC2",
          "ELB"
        ]
      },
      "ASGIAMInstanceProfile": {
        "description": "The IAM instance profile for the Auto Scaling group. EC2
instances launched with an IAM role automatically have AWS security credentials
available.",
        "type": "string",
        "default": "customer-mc-ec2-instance-profile"
      },
      "ASGInstanceDetailedMonitoring": {
        "description": "True to enable detailed monitoring on the instances in
the Auto Scaling group, false to use only basic monitoring. EC2 detailed monitoring
provides more frequent metrics, published at one-minute intervals, instead of the
five-minute intervals used in Amazon EC2 basic monitoring; it also incurs charges..",
        "type": "boolean",
        "default": true
      },
      "ASGInstanceRootVolumeIops": {
        "description": "The Iops to use for the root volume if io1 volume type is
specified.",
        "type": "integer",
        "minimum": 0,
        "maximum": 20000,
        "default": 0
```

```
      },
      "ASGInstanceRootVolumeName": {
        "description": "The name of the root volume to use. Defaults to the root
device name of the AMI.",
        "type": "string"
      },
      "ASGInstanceRootVolumeSize": {
        "description": "The size of the root volume for the instance. Defaults to
20 GiB for Linux and 60 GiB for Windows or the AMI root volume size, whichever is
higher.",
        "type": "integer",
        "minimum": 8,
        "maximum": 16000
      },
      "ASGInstanceRootVolumeType": {
        "description": "Choose io1 or gp2 for SSD-backed volumes optimized for
transactional workloads; choose standard for HDD-backed volumes optimized for large
streaming workloads.",
        "type": "string",
        "enum": [
          "standard",
          "io1",
          "gp2",
          "gp3"
        ],
        "default": "standard"
      },
      "ASGInstanceType": {
        "description": "The instance type for the Auto Scaling group to use when
creating new EC2 instances.",
        "type": "string",
        "default": "m5.large"
      },
      "ASGLoadBalancerNames": {
        "description": "A list of load balancers to associate with this Auto Scaling
group. Specify this if you want to place your Auto Scaling group behind a load
balancer.",
        "type": "array",
        "items": {
          "type": "string"
        },
        "minItems": 1,
        "maxItems": 10,
        "uniqueItems": true
```

```
      },
      "ASGMaxInstances": {
        "description": "The maximum number of instances you want in the Auto Scaling
group at any time. Defaults to 1 if not specified.",
        "type": "integer",
        "minimum": 1,
        "maximum": 1000,
        "default": 1
      },
      "ASGMinInstances": {
        "description": "The minimum number of instances you want in the Auto Scaling
group at any time. Defaults to 1 if not specified.",
        "type": "integer",
        "minimum": 1,
        "maximum": 1000,
        "default": 1
      },
      "ASGScaleDownMetricName": {
        "description": "The metric to use to in a scale-down event. Exceeding the
metric triggers an alarm.",
        "type": "string",
        "enum": [
          "CPUCreditUsage",
          "CPUCreditBalance",
          "CPUUtilization",
          "DiskReadOps",
          "DiskWriteOps",
          "DiskReadBytes",
          "DiskWriteBytes",
          "NetworkIn",
          "NetworkOut",
          "StatusCheckFailed",
          "StatusCheckFailed_Instance",
          "StatusCheckFailed_System"
        ],
        "default": "CPUUtilization"
      },
      "ASGScaleDownPolicyCooldown": {
        "description": "The number of seconds after a scale-down activity is
completed before any further scaling activities can start.",
        "type": "integer",
        "minimum": 120,
        "maximum": 600,
        "default": 300
```

```
      },
      "ASGScaleDownPolicyEvaluationPeriods": {
        "description": "The number of periods over which data is compared to the
specified ASGScaleDownMetricName threshold.",
        "type": "integer",
        "minimum": 2,
        "default": 4
      },
      "ASGScaleDownPolicyPeriod": {
        "description": "The time over which the specified ASGScaleDownPolicyStatistic
is applied. You must specify a time in seconds that is a multiple of 60.",
        "type": "integer",
        "multipleOf": 60,
        "minimum": 60,
        "default": 60
      },
      "ASGScaleDownPolicyScalingAdjustment": {
        "description": "The number of instances by which to scale down.",
        "type": "integer",
        "maximum": 0,
        "default": -1
      },
      "ASGScaleDownPolicyStatistic": {
        "description": "The statistic to apply to the alarm's
ASGScaleDownMetricName.",
        "type": "string",
        "enum": [
          "SampleCount",
          "Average",
          "Sum",
          "Minimum",
          "Maximum"
        ],
        "default": "Average"
      },
      "ASGScaleDownPolicyThreshold": {
        "description": "The value against which the specified
ASGScaleDownPolicyStatistic is compared.",
        "type": "number",
        "default": 35
      },
      "ASGScaleUpMetricName": {
        "description": "The metric to use in a scale-up event. Exceeding the metric
triggers an alarm.",
```

```
          "type": "string",
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
            "StatusCheckFailed_System"
          ],
          "default": "CPUUtilization"
        },
        "ASGScaleUpPolicyCooldown": {
          "description": "The amount of time, in seconds, after a scale-up activity is
completed before any further trigger-related scaling activities can start.",
          "type": "integer",
          "minimum": 60,
          "default": 60
        },
        "ASGScaleUpPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ASGScaleUpMetricName threshold.",
          "type": "integer",
          "minimum": 2,
          "default": 2
        },
        "ASGScaleUpPolicyPeriod": {
          "description": "The time over which the specified ASGScaleUpPolicyStatistic
is applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60,
          "default": 60
        },
        "ASGScaleUpPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale up.",
          "type": "integer",
          "minimum": 0,
          "default": 2
```

```
        },
        "ASGScaleUpPolicyStatistic": {
          "description": "The statistic to apply to the alarm's
ASGScaleUpMetricName.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ],
          "default": "Average"
        },
        "ASGScaleUpPolicyThreshold": {
          "description": "The value against which the specified
ASGScaleUpPolicyStatistic is compared.",
          "type": "number",
          "default": 75
        },
        "ASGSubnetIds": {
          "description": "One or more subnets for the Auto Scaling group to launch
instances into (scale up) or remove instances from (scale down), in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 2,
          "uniqueItems": true
        },
        "ASGUserData": {
          "description": "A newline-delimited string where each line is part of the
script to be run on boot.",
          "type": "string",
          "maxLength": 4096,
          "default": ""
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
```

```
          "ASGAmiId",
          "ASGInstanceType",
          "ASGInstanceRootVolumeName",
          "ASGInstanceRootVolumeType",
          "ASGInstanceRootVolumeSize",
          "ASGInstanceRootVolumeIops",
          "ASGIAMInstanceProfile",
          "ASGMinInstances",
          "ASGMaxInstances",
          "ASGDesiredCapacity",
          "ASGSubnetIds",
          "ASGEBSOptimized",
          "ASGLoadBalancerNames",
          "ASGUserData",
          "ASGCooldown",
          "ASGHealthCheckGracePeriod",
          "ASGHealthCheckType",
          "ASGInstanceDetailedMonitoring",
          "ASGScaleUpMetricName",
          "ASGScaleUpPolicyCooldown",
          "ASGScaleUpPolicyEvaluationPeriods",
          "ASGScaleUpPolicyPeriod",
          "ASGScaleUpPolicyScalingAdjustment",
          "ASGScaleUpPolicyStatistic",
          "ASGScaleUpPolicyThreshold",
          "ASGScaleDownMetricName",
          "ASGScaleDownPolicyCooldown",
          "ASGScaleDownPolicyEvaluationPeriods",
          "ASGScaleDownPolicyPeriod",
          "ASGScaleDownPolicyScalingAdjustment",
          "ASGScaleDownPolicyStatistic",
          "ASGScaleDownPolicyThreshold"
        ]
      },
      "required": [
        "ASGAmiId",
        "ASGSubnetIds"
      ]
    },
    "EnforceIMDSv2": {
      "description": "For the instance to be launched with only Instance Metadata
Service Version 2 (IMDSv2), use required; if IMDSv2 is not required, use optional.
Default is required.",
      "type": "string",
```

```
      "default": "required"
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "StackTemplateId",
      "Parameters",
      "TimeoutInMinutes",
      "Tags",
      "EnforceIMDSv2"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2u5rcyv5h34zn

Classifications:

- Management | Advanced stack components | RDS snapshot | Share

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Share RDS DB Snapshot",
  "description": "Share a snapshot of an Amazon Relational Database Service (RDS)
 database (DB) instance with another AMS account. Only snapshots encrypted with managed
 KMS keys can be shared.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ShareDBSnapshot.",
```

```
      "type": "string",
      "enum": [
        "AWSManagedServices-ShareDBSnapshot"
      ],
      "default": "AWSManagedServices-ShareDBSnapshot"
    },
    "Region": {
      "description": "The AWS Region where the DB snapshot is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBSnapshotName": {
          "description": "The DB snapshot name. Find this in the RDS console for that
 RDS DB.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "AccountId": {
          "description": "The ID of the AWS account the DB snapshots will be shared
 with, in the form 123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "DBSnapshotName",
          "AccountId"
        ]
      },
```

```
      "additionalProperties": false,
      "required": [
        "DBSnapshotName",
        "AccountId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2uimt36z7j6vn

Classifications:

- Management | Advanced stack components | RDS database stack | Restore to point in time

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Restore RDS DB Instance To Point In Time",
  "description": "Restore an RDS DB instance to a point in time.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RestoreRDSInstanceToPointInTime.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RestoreRDSInstanceToPointInTime"
      ],
      "default": "AWSManagedServices-RestoreRDSInstanceToPointInTime"
    },
```

```
   "Region": {
     "description": "The AWS Region in which the AWS resource is located, in the form
us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "SourceDBInstanceIdentifier": {
         "description": "Identifier of the source DB instance to restore to a point in
time.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^[a-z](?!.*--)(?!.*-$)[a-z0-9-]{0,62}$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "TargetDBInstanceIdentifier": {
         "description": "A meaningful name for the new DB instance.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^[a-z](?!.*--)(?!.*-$)[a-z0-9-]{0,62}$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "RestoreTime": {
         "description": "Date and time to restore from in Universal Coordinated Time
(UTC) format, for example 2009-09-07T23:45:00Z. Leave empty to restore the DB instance
from the latest backup time.",
         "type": "array",
         "items": {
           "type": "string",
           "default": "",
           "pattern": "^20\\d{2}-\\d{2}-\\d{2}T\\d{2}:\\d{2}:\\d{2}Z$|^$"
         },
         "minItems": 1,
         "maxItems": 1
       },
       "DBInstanceClass": {
```

```
        "description": "The compute and memory capacity for the DB instance. Leave
empty to use the same instance class as the source DB instance.",
        "type": "array",
        "items": {
          "type": "string",
          "default": "",
          "pattern": "^db\\.[a-z0-9]+\\.[a-z0-9]+$|^$"
        },
        "minItems": 1,
        "maxItems": 1
      },
      "DBOptionGroupName": {
        "description": "The option group that this DB instance is associated with.
If none is provided, the default option group is associated. An option group can
specify features, called options, that are available for a particular Amazon RDS DB
instance.",
        "type": "array",
        "items": {
          "type": "string",
          "default": "",
          "pattern": "^[a-zA-Z](?!.*--)[a-z0-9-]{1,255}[^-]$|^$"
        },
        "minItems": 0,
        "maxItems": 1
      },
      "DBParameterGroupName": {
        "description": "The name of an existing DB parameter group. If none is
provided, the default parameter group is associated. A DB parameter group acts
as a container for engine configuration values that are applied to one or more DB
instances.",
        "type": "array",
        "items": {
          "type": "string",
          "default": "",
          "pattern": "^[a-zA-Z](?!.*--)[a-z0-9-]{1,255}[^-]$|^$"
        },
        "minItems": 0,
        "maxItems": 1
      }
    },
    "metadata": {
      "ui:order": [
        "SourceDBInstanceIdentifier",
        "TargetDBInstanceIdentifier",
```

```
          "RestoreTime",
          "DBInstanceClass",
          "DBOptionGroupName",
          "DBParameterGroupName"
        ]
      },
      "required": [
        "SourceDBInstanceIdentifier",
        "TargetDBInstanceIdentifier"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2utx36abv83pv

Classifications:

- Management | Patching | Patch window | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Maintenance Window",
  "description": "Modify patch maintenance window settings created using version 1 of
 change type ct-0el2j07llrxs7.",
  "type": "object",
  "properties": {
    "DocumentName": {
```

```
      "description": "Must be AWSManagedServices-UpdateMaintenanceWindow.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateMaintenanceWindow"
      ],
      "default": "AWSManagedServices-UpdateMaintenanceWindow"
    },
    "Region": {
      "description": "The AWS Region where the SSM maintenance window is located, in
the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "WindowId": {
          "description": "The ID of the maintenance window (for example,
mw-012345678910abcef).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^mw-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "OnlyCheckForMaintenanceWindowDrift": {
          "description": "True to generate a drift detection report (visible in the
output section of the executed RFC). False to not generate a drift detection report.
If the request has mutable changes, such as modifying maintenance window settings, use
False.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "True",
              "False"
            ],
            "default": "False"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
      "EmailAction": {
        "description": "Add the specified NotificationEmails to the patch maintenance
window with 'Add', remove them from the window with 'Remove'. If you have no
NotificationEmails, use 'None'.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "None",
            "Add",
            "Remove"
          ]
        },
        "minItems": 1,
        "maxItems": 1
      },
      "BypassDriftDetection": {
        "description": "True to bypass checks preventing introduction of drift in
CloudFormation resources. If the request should not generate drift, use False.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "True",
            "False"
          ],
          "default": "False"
        },
        "minItems": 1,
        "maxItems": 1
      },
      "NotificationEmails": {
        "description": "Up to four email addresses, in a comma separated list.
Specify that they be added, or removed, from the provided maintenance window with the
EmailAction parameter.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^$|([a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+)+.*",
          "default": ""
        },
        "minItems": 0,
        "maxItems": 4,
        "uniqueItems": true
```

```
        },
        "Duration": {
          "description": "The duration of the maintenance window in hours.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0]$|^(2[0-4]|1[0-9]|[1-9])$",
            "default": "0"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "PatchGroupName": {
          "description": "A new name for the patch group for the maintenance window
to target. Target EC2 instances must be tagged with the tag key \"Patch Group\" and
the tag value defined by this parameter. For example, provided the name MyApp, the
maintenance window targets any EC2 instances with the tag key \"Patch Group\" and the
tag value \"MyApp\". To keep the current patch group name, leave blank.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^[a-zA-Z0-9_\\-.]{3,128}$",
            "default": ""
          },
          "minItems": 0,
          "maxItems": 1
        },
        "Schedule": {
          "description": "The schedule of the maintenance window in the form of a cron
or rate expression.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^.{0,256}$",
            "default": ""
          },
          "minItems": 0,
          "maxItems": 1
        },
        "ScheduleTimezone": {
          "description": "The time zone that the scheduled maintenance window
executions are based on, in Internet Assigned Numbers Authority (IANA) format.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^$|(^[a-zA-Z_]+(\\\\+|/)?[a-zA-Z0-9_-]*(\\\\+|/)?[a-zA-Z0-9_-]+
$)",
            "default": ""
          },
          "minItems": 0,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "WindowId",
          "OnlyCheckForMaintenanceWindowDrift",
          "EmailAction",
          "BypassDriftDetection",
          "NotificationEmails",
          "Duration",
          "PatchGroupName",
          "Schedule",
          "ScheduleTimezone"
        ]
      },
      "additionalProperties": false,
      "required": [
        "EmailAction",
        "OnlyCheckForMaintenanceWindowDrift",
        "WindowId",
        "BypassDriftDetection"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
```

```
}
```

Schema for Change Type ct-2uw99b8hpncnu

Classifications:

- Deployment | Advanced stack components | Elastic File System (EFS) | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EFS stack",
  "description": "Use to create a Elastic File System (EFS) stack",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 50 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
```

```
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,255}$",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "Encrypted": {
          "description": "True to create an encrypted file system, false to create a
file system that is not encrypted.",
          "type": "boolean",
          "default": true
        },
```

```
        "KmsKeyId": {
          "description": "The AWS Key Management Service (AWS KMS) customer master key
 (CMK) ID to use for the encrypted file system if Encrypted = true. If not specified,
 the default CMK for Amazon EFS is used.",
          "type": "string",
          "maxLength": 2048
        },
        "PerformanceMode": {
          "description": "The performance mode of the file system. We recommend
 generalPurpose for most file systems.",
          "type": "string",
          "enum": [
            "generalPurpose",
            "maxIO"
          ],
          "default": "generalPurpose"
        },
        "MountTargets": {
          "description": "Specifications for the file system mount targets.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "AvailabilityZone": {
                "description": "The availability zone for the mount target. Only one
 mount target per availability zone is required.",
                "type": "string",
                "pattern": "^[a-z0-9-.]{1,127}$"
              },
              "SubnetId": {
                "description": "The ID of a subnet in the specified mount target
 availability zone, in the form subnet-0123abcd or subnet-01234567890abcdef. If not
 specified, a random subnet in the availability zone is chosen.",
                "type": "string",
                "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
              },
              "IpAddress": {
                "description": "An IPv4 address that is within the address range of the
 specified SubnetId property. If not specified, Amazon EFS assigns an address that is
 within the range of the specified subnet.",
                "type": "string",
                "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}
([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])$"
              }
```

```
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "AvailabilityZone",
                "SubnetId",
                "IpAddress"
              ]
            },
            "required": [
              "AvailabilityZone"
            ]
          },
          "minItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Encrypted",
          "KmsKeyId",
          "PerformanceMode",
          "MountTargets"
        ]
      },
      "required": [
        "Encrypted",
        "PerformanceMode",
        "MountTargets"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "Tags"
    ]
  },
  "required": [
```

```
    "Name",
    "Description",
    "VpcId",
    "Parameters",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-2uzbqr7x7mekd

Classifications:

- Management | Standard stacks | Stack | Update termination protection

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Termination Protection",
  "description": "Update existing defined termination protection for stacks.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
ManageResourceTerminationProtection.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ManageResourceTerminationProtection"
      ],
      "default": "AWSManagedServices-ManageResourceTerminationProtection"
    },
    "Region": {
      "description": "The AWS Region in which the stack is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ResourceId": {
          "description": "Stack name.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^stack-[a-zA-Z0-9\\-]{1,122}$"
          },
          "maxItems": 1
        },
        "TerminationProtectionDesiredState": {
          "description": "Enabled to protect your stack against elimination. Disabled
 to allow your stack to be eliminated.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "enabled",
              "disabled"
            ]
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ResourceId",
          "TerminationProtectionDesiredState"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ResourceId",
        "TerminationProtectionDesiredState"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-2v82sp4np40ki

Classifications:

- Management | Advanced stack components | Target group | Update (for ALB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update target group for ALB",
  "description": "Use to update properties of an existing Target Group for an
 Application Load Balancer created by CT id ct-1r19m51jeijlk.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the Target Group (for ALB) that you are updating,
 in the form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "HealthCheckHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$"
        },
        "HealthCheckUnhealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check failure required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$"
```

```
        },
        "HealthCheckInterval": {
          "type": "integer",
          "description": "The approximate interval, in seconds, between health checks.
 The supported values are 5 seconds to 300 seconds.",
          "minimum": 5,
          "maximum": 300
        },
        "HealthCheckTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, to wait for a response to
 a health check. Must be less than the value for HealthCheckInterval. The supported
 values are 2 seconds to 60 seconds.",
          "pattern": "60|[1-5]{1}[0-9]{1}|[2-9]{1}|^$"
        },
        "HealthCheckTargetPath": {
          "type": "string",
          "description": "The ping path destination on the application hosts where the
 load balancer sends health check requests."
        },
        "HealthCheckTargetPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health checks
 on targets. The default is traffic-port, which indicates the port on which each target
 receives traffic from the load balancer.",
          "pattern": "[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2]
[0-9]|6553[0-5]|traffic-port|"
        },
        "HealthCheckTargetProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS"
          ]
        },
        "ValidHTTPCode": {
          "type": "string",
          "description": "The HTTP codes that a healthy target application server must
 use in response to a health check. You can specify multiple values such as 200,202, or
 a range of values such as 200-499. Only applicable if HealthCheckTargetProtocol = HTTP
 or HTTPS.",
          "pattern": "^$|([2-4]{1}[0-9]{2}($|-|,))+"
```

```
        },
        "DeregistrationDelayTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing to
 wait before changing the state of a deregistering target from draining to unused.",
          "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})"
        },
        "SlowStartDuration": {
          "type": "string",
          "description": "The time period, in seconds, during which the load balancer
 sends a newly registered target a linearly-increasing share of the target group
 traffic.",
          "pattern": "[3-9]{1}[0-9]{1}|[1-8]{1}[0-9]{2}|900|0|"
        },
        "StickinessCookieExpirationPeriod": {
          "type": "string",
          "description": "The time period, in seconds, after which the cookie is
 considered stale. If this parameter isn't specified, the sticky session lasts for the
 duration of the browser session.",
          "pattern": "[1-9]{1}[0-9]{0,4}|[1-5]{1}[0-9]{5}|60[0-3]{1}[0-9]{3}|604[0-7]
{1}[0-9]{2}|604800|"
        },
        "Target1ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target1Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target1AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target1ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
```

```
            "all"
          ]
        },
        "Target2ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target2ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target3ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target3AvailabilityZone": {
```

```
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target3ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target4ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target4AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target4ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target5ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target5Port": {
          "type": "string",
```

```
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target5AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target5ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target6ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target6Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target6AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target6ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target7ID": {
          "type": "string",
```

```
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target7Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target7AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target7ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
            "",
            "all"
          ]
        },
        "Target8ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}"
        },
        "Target8Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]"
        },
        "Target8AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType
 = ip, and the IP address in Target8ID is outside the VPC, use all. Otherwise, leave
 blank.",
          "enum": [
```

```
            "",
            "all"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "DeregistrationDelayTimeout",
          "SlowStartDuration",
          "StickinessCookieExpirationPeriod",
          "HealthCheckTargetPath",
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckUnhealthyThreshold",
          "HealthCheckInterval",
          "HealthCheckTimeout",
          "ValidHTTPCode",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
          "Target4AvailabilityZone",
          "Target5ID",
          "Target5Port",
          "Target5AvailabilityZone",
          "Target6ID",
          "Target6Port",
          "Target6AvailabilityZone",
          "Target7ID",
          "Target7Port",
          "Target7AvailabilityZone",
          "Target8ID",
          "Target8Port",
          "Target8AvailabilityZone"
        ]
      },
```

```
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2w3rbmnny1qpo

Classifications:

- Management | Directory Service | DNS | Add A record

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add DNS A Record",
  "description": "Add a new static DNS A record in AWS Managed Microsoft Active
 Directory (AD). For multi-account landing zone (MALZ), use this change type in the
 shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-CreateDNSARecord-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateDNSARecord-Admin"
      ],
      "default": "AWSManagedServices-CreateDNSARecord-Admin"
    },
    "Region": {
```

```
      "description": "The AWS Region where AWS managed Microsoft AD in Directory
 Service is located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RecordName": {
          "description": "A meaningful name for the DNS A record.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-\\_]{1,63}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IPAddress": {
          "description": "The IPv4 address the DNS A record resolves to.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(^(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\\.){3}(?:25[0-5]|
2[0-4][0-9]|[01]?[0-9][0-9]?)(, )?){1,6}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "TTLValue": {
          "description": "The Time to Live (TTL) value in format hh:mm:ss for a DNS
 resource record (default is 01:00:00).",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?:(?:([01]?\\d|2[0-3]):)?([0-5]?\\d):)?([0-5]?\\d)$",
            "default": "01:00:00"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
```

```
          "RecordName",
          "IPAddress",
          "TTLValue"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RecordName",
        "IPAddress"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2wlfo2jxj2rkj

Classifications:

- Management | Managed landing zone | Application account | Confirm offboarding

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Confirm Account Offboarding",
  "description": "Confirm offboarding of the specified application account. Run this
 from the application account that you want offboarded. Once confirmed, run the Execute
 offboarding CT (ct-0vdiy51oyrhhm) from the associated management account. Note that
 this offboarding is intended for account closure and cannot be undone",
  "type": "object",
  "properties": {
```

```
    "RequestType": {
      "description": "Must be OffboardingConfirmation.",
      "type": "string",
      "enum": [
        "OffboardingConfirmation"
      ],
      "default": "OffboardingConfirmation"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AccountId": {
          "description": "The unique identifier (ID) of the application account to
offboard.",
          "type": "string",
          "pattern": "^[0-9]{12}$"
        },
        "AccountEmail": {
          "description": "The email associated with the application account to
offboard.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AccountId",
          "AccountEmail"
        ]
      },
      "required": [
        "AccountId",
        "AccountEmail"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Parameters",
      "RequestType"
    ]
  },
  "additionalProperties": false,
```

```
  "required": [
    "Parameters",
    "RequestType"
  ]
}
```

Schema for Change Type ct-2wllq61djysxz

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create from backup (for Aurora)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS Aurora Stack From Backup",
  "description": "Create an AWS Relational Database Service (RDS) Aurora stack from AWS
 Backup.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
```

```
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-j24cifrdi0untnsn6",
      "type": "string",
      "enum": [
        "stm-j24cifrdi0untnsn6"
      ],
      "default": "stm-j24cifrdi0untnsn6"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
```

```
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SnapshotIdentifier": {
          "type": "string",
          "description": "The identifier for the DB snapshot or DB cluster snapshot to
restore from.",
          "pattern": "^[a-zA-Z][a-zA-Z0-9-:]{1,255}$"
        },
        "AutoMinorVersionUpgrade": {
          "type": "string",
          "description": "True if the RDS instance should have automatic minor version
upgrade, false if it should not. Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "BackupRetentionPeriod": {
          "type": "integer",
          "description": "The number of days for which automatic database (DB)
snapshots are retained. Range is 1 - 35.",
          "default": 7,
          "minimum": 1,
          "maximum": 35
        },
        "ClusterName": {
          "type": "string",
          "description": "Optional identifier for the DB Cluster that is created with
your instance. If you do not provide one, a default identifier based on the instance
identifier is used. The cluster identifier is used in determining the cluster's
connection endpoint.",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$",
          "default": ""
        },
        "DBEngine": {
          "type": "string",
          "description": "The name of the engine for the Aurora database. Not every
database engine is available for every AWS region. Engine compatability is determined
```

```
 by engine type (aurora=MySQL 5.6, aurora-mysql=MySQL 5.7, aurora-postgresql=PostgreSQL
 10.4, 9.6.9 or 9.6.8).",
          "enum": [
            "aurora",
            "aurora-mysql",
            "aurora-postgresql"
          ],
          "default": "aurora"
        },
        "DBName": {
          "type": "string",
          "description": "A name for the database. The meaning of this parameter
 differs according to the database engine you use.",
          "pattern": "^$|[a-zA-Z0-9]{1,64}$",
          "default": ""
        },
        "DBClusterParameterGroupName": {
          "description": "The name of an existing DB cluster parameter group. The
 parameter group must be compatible with the DBEngine and the EngineVersion.",
          "type": "string",
          "pattern": "^(?!.*--.*)(?!.*-$)[a-zA-Z][a-zA-Z0-9-.]{0,254}$"
        },
        "DBSubnetGroupName": {
          "type": "string",
          "description": "The name of an existing DB subnet group provisioned with the
 \"RDS database stack | Create DB subnet group\" change type.",
          "pattern": "^[a-zA-Z0-9._-]{1,255}$"
        },
        "EngineVersion": {
          "type": "string",
          "description": "The version number of the database engine to use. Not every
 database version is available for every AWS region.",
          "pattern": "^\\d.\\d.\\d{2}[a-z]$|^5.\\d.mysql_aurora.\\d.\\d{2}.\\d$|^8.\
\d.mysql_aurora.\\d.\\d{2}.\\d$|^(\\d{2}.\\d{0,2})$|^$",
          "default": ""
        },
        "InstanceType": {
          "type": "string",
          "description": "The instance type to use, this determines the compute and
 memory capacity for the DB instance. Not every instance type is available for every
 database engine.",
          "enum": [
            "db.serverless",
            "db.t2.small",
```

```
            "db.t2.medium",
            "db.t3.micro",
            "db.t3.small",
            "db.t3.medium",
            "db.t3.large",
            "db.t3.xlarge",
            "db.t3.2xlarge",
            "db.t4g.medium",
            "db.t4g.large",
            "db.r3.large",
            "db.r3.xlarge",
            "db.r3.2xlarge",
            "db.r3.4xlarge",
            "db.r3.8xlarge",
            "db.r4.large",
            "db.r4.xlarge",
            "db.r4.2xlarge",
            "db.r4.4xlarge",
            "db.r4.8xlarge",
            "db.r4.16xlarge",
            "db.r5.large",
            "db.r5.xlarge",
            "db.r5.2xlarge",
            "db.r5.4xlarge",
            "db.r5.8xlarge",
            "db.r5.12xlarge",
            "db.r5.16xlarge",
            "db.r5.24xlarge",
            "db.r6g.large",
            "db.r6g.xlarge",
            "db.r6g.2xlarge",
            "db.r6g.4xlarge",
            "db.r6g.8xlarge",
            "db.r6g.12xlarge",
            "db.r6g.16xlarge",
            "db.x2g.large",
            "db.x2g.xlarge",
            "db.x2g.2xlarge",
            "db.x2g.4xlarge",
            "db.x2g.8xlarge",
            "db.x2g.12xlarge",
            "db.x2g.16xlarge"
          ],
          "default": "db.r4.large"
```

```
        },
        "MultiAZ": {
          "type": "string",
          "description": "True to have a secondary replica of your DB instance created
 in another Availability Zone for failover support, false to not have a standby.
 Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "Port": {
          "type": "string",
          "description": "The port for the instance. Valid range is: 1150-65535.
 Specifying 0 assigns the default based on the selected DBEngine (aurora=3306, aurora-
mysql=3306, aurora-postgresql=5432).",
          "pattern": "^(0|11[5-8][0-9]|119[0-9]|1[2-9][0-9]{2}|[2-9][0-9]{3}|[1-5][0-9]
{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])$",
          "default": "0"
        },
        "PreferredBackupWindow": {
          "type": "string",
          "description": "The daily time range during which automated backups are
 created. Must be in the format hh:mm-hh:mm (24-hour format), in Universal Coordinated
 Time (UTC). Must not conflict with the PreferredMaintenanceWindow setting, and must be
 at least 30 minutes.",
          "pattern": "^[0-9]{2}:[0-9]{2}-[0-9]{2}:[0-9]{2}$",
          "default": "22:00-23:00"
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
          "description": "The weekly time range during which system maintenance
 can occur, in UTC. Must be in the format ddd:hh:mm-ddd:hh:mm (24-hour format), in
 Universal Coordinated Time (UTC) and must be at least 30 minutes. If you don't specify
 PreferredMaintenanceWindow, then Amazon RDS assigns a 30-minute maintenance window on
 a randomly selected day of the week.",
          "pattern": "^$|[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$",
          "default": ""
        },
        "ServerlessScalingMaxCapacity": {
          "description": "The maximum number of Aurora capacity units (ACUs) for a DB
 instance in an Aurora Serverless cluster. The largest value that you can use is 128.0.
 Only applies to db.serverless InstanceType.",
```

```
          "type": "number",
          "minimum": 1,
          "maximum": 128,
          "default": 1
        },
        "ServerlessScalingMinCapacity": {
          "description": "The minimum number of Aurora capacity units (ACUs) for a DB
instance in an Aurora Serverless cluster. The smallest  value that you can use is 0.5.
Only applies to db.serverless InstanceType.",
          "type": "number",
          "minimum": 0.5,
          "maximum": 128,
          "default": 0.5
        }
      },
      "metadata": {
        "ui:order": [
          "SnapshotIdentifier",
          "DBEngine",
          "EngineVersion",
          "InstanceType",
          "MultiAZ",
          "DBName",
          "ClusterName",
          "DBClusterParameterGroupName",
          "DBSubnetGroupName",
          "Port",
          "AutoMinorVersionUpgrade",
          "BackupRetentionPeriod",
          "PreferredBackupWindow",
          "PreferredMaintenanceWindow",
          "ServerlessScalingMaxCapacity",
          "ServerlessScalingMinCapacity"
        ]
      },
      "required": [
        "SnapshotIdentifier",
        "DBEngine",
        "EngineVersion",
        "DBSubnetGroupName"
      ],
      "additionalProperties": false
    }
  },
```

```
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2wrvu4kca9xky

Classifications:

- Management | AMS Resource Scheduler | State | Enable

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable AMS Resource Scheduler",
  "description": "Enable AMS Resource Scheduler in the account where it was previously
 disabled. This will re-enable scheduling of resources for automatic start or stop
 actions where the resources are already tagged with a valid schedule. Make sure to
 verify currently tagged resources and schedules before enabling the scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAMSResourceSchedulerStack-
Admin.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
      ],
      "default": "AWSManagedServices-HandleAMSResourceSchedulerStack-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SchedulingActive": {
          "description": "Specify the value: Yes. This explicitly requests that the
Resource Scheduler be enabled from a disabled state. Default is Yes.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "Yes"
            ],
            "default": "Yes"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Action": {
          "type": "string",
          "description": "(Required) The Action to be performed.",
          "enum": [
            "Update"
          ],
          "default": "Update"
        }
      },
      "metadata": {
        "ui:order": [
          "SchedulingActive",
          "Action"
        ]
      },
      "required": [
        "SchedulingActive",
```

```
        "Action"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2x14cv67uym46

Classifications:

- Management | Advanced stack components | Bastions | Update instance size (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Instance Size",
  "description": "Update the instance size for an RDP or SSH customer bastion in an AMS
 account.",
  "type": "object",
  "properties": {
    "BastionType": {
      "description": "The bastion type to update.",
      "type": "string",
      "default": "RDP Bastion",
      "enum": [
        "RDP Bastion",
        "SSH Bastion"
      ]
    },
```

```
    "InstanceType": {
      "description": "The new instance type for the bastion. If BastionType = SSH
 Bastion, the minimum instance size is t3.small. If BastionType = RDP Bastion, the
 minimum instance size is t3.medium.",
      "type": "string",
      "pattern": "^[a-z0-9]+\\.[a-z0-9]+$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "BastionType",
      "InstanceType",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "BastionType",
    "InstanceType"
  ]
}
```

Schema for Change Type ct-2xd2anlb5hbzo

Classifications:

- Management | Directory Service | Directory | Unshare directory

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Unshare Directory",
```

```
  "description": "Stops the directory sharing between the directory owner and consumer
 accounts. Run this in your Shared Service account that has Managed Active Directory.
 This change type is only supported for multi-account landing zone (MALZ).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-UnshareDirectory.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UnshareDirectory"
      ],
      "default": "AWSManagedServices-UnshareDirectory"
    },
    "Region": {
      "description": "The AWS Region where the directory is located, in the form of us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DirectoryId": {
          "description": "The identifier of the AWS Managed Microsoft Active directory
 that you want to stop sharing.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^d-[0-9a-f]{10}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "UnshareTarget": {
          "description": "Identifier for the directory consumer account to unshare the
 directory from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "DirectoryId",
          "UnshareTarget"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DirectoryId",
        "UnshareTarget"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2y6q4vco4miyp

Classifications:

- Management | Advanced stack components | EBS Volume | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update EBS volumes.",
  "description": "Modify the properties of an existing Elastic Block Store (EBS) volume
 stack created using CT id ct-16xg8qguovg2w, version 1.0. No service interruption is
 expected during the update.",
  "type": "object",
```

```
  "properties": {
    "VpcId": {
      "description": "ID of the VPC that contains the EC2 instance the EBS volumes are
 attached to, in the form vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "ID of the stack instance that contains the EBS Volumes, in the
 form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Volume1Iops": {
          "type": "string",
          "description": "The Iops to use for Volume1 if Volume1Type is io1, io2 or
 gp3. If Volume1Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume1Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume1Size": {
          "type": "string",
          "description": "The size for Volume1 in GiB. The size can be increased, but
 not decreased.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume1Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume1 if Volume1Type is gp3. If
 Volume1Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume1Type": {
          "type": "string",
          "description": "The volume type for Volume1. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
```

```
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ]
        },
        "Volume2Iops": {
          "type": "string",
          "description": "The Iops to use for Volume2 if Volume2Type is io1, io2 or
 gp3. If Volume2Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume2Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume2Size": {
          "type": "string",
          "description": "The size for Volume2 in GiB. The size can be increased, but
 not decreased.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume2Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume2 if Volume2Type is gp3. If
 Volume2Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume2Type": {
          "type": "string",
          "description": "The volume type for Volume2. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
```

```
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ]
        },
        "Volume3Iops": {
          "type": "string",
          "description": "The Iops to use for Volume3 if Volume3Type is io1, io2 or
 gp3. If Volume3Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume3Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume3Size": {
          "type": "string",
          "description": "The size for Volume3 in GiB. The size can be increased, but
 not decreased.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume3Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume3 if Volume3Type is gp3. If
 Volume3Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume3Type": {
          "type": "string",
          "description": "The volume type for Volume3. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
```

```
          ]
        },
        "Volume4Iops": {
          "type": "string",
          "description": "The Iops to use for Volume4 if Volume4Type is io1, io2 or
 gp3. If Volume4Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume4Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume4Size": {
          "type": "string",
          "description": "The size for Volume4 in GiB. The size can be increased, but
 not decreased.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume4Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume4 if Volume4Type is gp3. If
 Volume4Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume4Type": {
          "type": "string",
          "description": "The volume type for Volume4. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ]
        },
        "Volume5Iops": {
          "type": "string",
```

```
          "description": "The Iops to use for Volume5 if Volume5Type is io1, io2 or
 gp3. If Volume5Type is not io1, io2 or gp3, any value provided here is ignored. If
 Volume5Type is gp3, then the Iops should be between 3000 and 16000, else it should be
 between 100 and 64000.",
          "pattern": "^$|^([1-9][0-9]{2}|[1-9][0-9]{3}|[1-5][0-9][0-9]{3}|[6][0-3][0-9]
{3}|64000)$"
        },
        "Volume5Size": {
          "type": "string",
          "description": "The size for Volume5 in GiB. The size can be increased, but
 not decreased.",
          "pattern": "^([1-9]|[1-9][0-9]{1}|[1-9][0-9]{2}|[1-9][0-9]{3}|[1][0-5][0-9]
{3}||[1][6][0-3][0-8][0-4]|16384)$"
        },
        "Volume5Throughput": {
          "type": "string",
          "description": "The Throughput to use for Volume5 if Volume5Type is gp3. If
 Volume5Type is not gp3, any value provided here is ignored. The Throughput should be
 between 125 and 1000. Default is 125.",
          "pattern": "^$|^([1][2][5-9]$|[1][3-9][0-9]$|[2-9][0-9][0-9]$|1000)$"
        },
        "Volume5Type": {
          "type": "string",
          "description": "The volume type for Volume5. Choose io1, io2, gp2 or gp3 for
 SSD-backed volumes optimized for transactional workloads. Choose sc1 or st1 for HDD-
backed volumes optimized for large streaming workloads. Choose standard for HDD-backed
 volumes suitable for workloads where data is infrequently accessed.",
          "enum": [
            "io1",
            "io2",
            "gp2",
            "gp3",
            "sc1",
            "st1",
            "standard"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "Volume1Size",
          "Volume1Type",
          "Volume1Iops",
          "Volume1Throughput",
```

```
          "Volume2Size",
          "Volume2Type",
          "Volume2Iops",
          "Volume2Throughput",
          "Volume3Size",
          "Volume3Type",
          "Volume3Iops",
          "Volume3Throughput",
          "Volume4Size",
          "Volume4Type",
          "Volume4Iops",
          "Volume4Throughput",
          "Volume5Size",
          "Volume5Type",
          "Volume5Iops",
          "Volume5Throughput"
        ]
      }
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-2yja7ihh30ply

Classifications:

- Management | AWS Backup | Backup plan | Enable cross account copy (Management account)

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable cross account copy (Management account)",
  "description": "Enable and configure cross-account backup and monitoring in a
 management account. This automation can only be completed successfully in a management
 account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
HandleConfigureCrossAccountBackupInManagementAccount-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleConfigureCrossAccountBackupInManagementAccount-
Admin"
      ],
      "default": "AWSManagedServices-
HandleConfigureCrossAccountBackupInManagementAccount-Admin"
    },
    "Region": {
      "description": "The AWS Region to enable the cross account backup and monitoring
 in, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DestinationAccountId": {
          "description": "The destination account ID of the cross-account backup to
 enable.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "maxItems": 1
        },
        "SourceAccountId": {
          "description": "The source account ID of the cross-account backup to
 enable.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
```

```
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SourceAccountId",
          "DestinationAccountId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DestinationAccountId",
        "SourceAccountId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2z60dyvto9g6c

Classifications:

- Deployment | Advanced stack components | RDS database stack | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS database stack",
```

```
 "description": "Create an Amazon Relational Database Service (RDS) DB instance. To
provision an Aurora single instance or multi-AZ instances, use CT ct-2jvzjwunghrhy.",
 "type": "object",
 "properties": {
   "Description": {
     "description": "Meaningful information about the resource to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "VpcId": {
     "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
     "type": "string",
     "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
   },
   "StackTemplateId": {
     "description": "Must be stm-sl81ze20000000000.",
     "type": "string",
     "enum": [
       "stm-sl81ze20000000000"
     ]
   },
   "Name": {
     "description": "A name for the stack or stack component; this becomes the Stack
Name.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "minLength": 1,
```

```
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "description": "Specifications for the stack.",
      "type": "object",
      "properties": {
        "RDSAllocatedStorage": {
          "description": "The size of the database in gigabytes (GB). The acceptable
limits for this value relate to the engine and storage type that you specify. For
details, see AWS documentation on DB instance storage.",
          "type": "number"
        },
        "RDSAutoMinorVersionUpgrade": {
          "description": "True to apply minor engine upgrades automatically to the DB
instance during the maintenance window, false to not. Default is false.",
          "type": "boolean",
          "default": false
        },
```

```
      "RDSBackupRetentionPeriod": {
        "description": "The number of days for which automatic DB snapshots are
retained. Setting this to a positive number enables backups. Setting this to 0
disables automated backups.",
        "type": "number",
        "minimum": 0,
        "maximum": 35,
        "default": 7
      },
      "RDSCharacterSetName": {
        "description": "The character set to associate with the DB instance. This
is applicable only if RDSDBEngine = oracle-se, oracle-se1, oracle-se2, oracle-ee,
sqlserver-ee, sqlserver-se, sqlserver-ex or sqlserver-web.",
        "type": "string",
        "default": ""
      },
      "RDSDBEngine": {
        "description": "The name of the database engine for the DB instance. Not
every database engine is available for every AWS region.",
        "type": "string",
        "enum": [
          "mariadb",
          "mysql",
          "oracle-se2",
          "oracle-se1",
          "oracle-se",
          "oracle-ee",
          "sqlserver-ee",
          "sqlserver-se",
          "sqlserver-ex",
          "sqlserver-web",
          "postgres"
        ]
      },
      "RDSDBInstanceIdentifier": {
        "description": "A name for the DB instance. It must begin with a letter, must
contain only letters, digits and hyphens and must not end with a hyphen or contain two
consecutive hyphens. If left blank AWS CloudFormation generates a unique physical ID
and uses that ID for the DB instance.",
        "type": "string",
        "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$",
        "minLength": 0,
        "maxLength": 63
      },
```

```
      "RDSDBName": {
        "description": "A name for the database. The meaning of this parameter
differs according to the database engine you use.For example, the name can't be longer
than 8 characters for Oracle database, for some others the DBName must begin with a
letter and contain only alphanumeric characters. For details on DBName requirements,
see the AWS RDS documentation for \"CreateDBInstance\" API.",
        "type": "string"
      },
      "RDSDBParameterGroupName": {
        "description": "The name of an existing DB parameter group.",
        "type": "string"
      },
      "RDSDeletionProtection": {
        "description": "True to enable DB instance deletion protection, false to not.
Default is false.",
        "type": "boolean",
        "default": false
      },
      "RDSEngineVersion": {
        "description": "The version number of the database engine to use; for
example, 5.7.30 for the MySQL engine. For details on engine versions, see the AWS RDS
documentation \"CreateDBInstance\" API.",
        "type": "string"
      },
      "RDSInstanceType": {
        "description": "The compute and memory capacity for the DB instance. Not all
DB instance classes are available in all AWS Regions, or for all database engines. For
details on the list of DB instance classes available for a specific engine, see the
AWS RDS documentation for \"CreateDBInstance\" API.",
        "type": "string",
        "pattern": "^db\\.[a-z0-9]+\\.[a-z0-9]+$",
        "default": "db.m4.large"
      },
      "RDSIOPS": {
        "description": "The provisioned IOPS for RDS storage. Must be a multiple
between 3 and 10 of the storage amount for the DB instance. Must also be an integer
multiple of 1000. For example, if the size of your DB instance is 500 GB, then your
Iops value can be 2000, 3000, 4000, or 5000.",
        "type": "number",
        "default": 0
      },
      "RDSLicenseModel": {
```

```
          "description": "License model information for the DB instance. This is
 applicable only if RDSDBEngine = oracle-se1 or oracle-se2. Default is license-
included.",
          "type": "string",
          "enum": [
            "bring-your-own-license",
            "license-included"
          ]
        },
        "RDSMasterUsername": {
          "description": "The name that you will use with the configured
 RDSMasterUserPassword to log in to your DB instance. Must begin with a letter
 and contain only alphanumeric characters. For details regarding DB engine related
 constraints on the user name, see the AWS RDS documentation for \"CreateDBInstance\"
 API.",
          "type": "string",
          "pattern": "^[a-zA-Z][a-zA-Z0-9]{1,128}$",
          "minLength": 1,
          "maxLength": 128
        },
        "RDSMasterUserPassword": {
          "description": "The password that you will use with the configured
 RDSMasterUserName to log in to your DB instance. Must contain from 8 to 30 printable
 ASCII characters (excluding backslash, double quotes, and at sign).",
          "type": "string",
          "pattern": "^[!#-.0-?A-~]{8,30}$",
          "metadata": {
            "ams:sensitive": true
          }
        },
        "RDSMultiAZ": {
          "description": "True to have a standby replica of your DB instance created in
 another Availability Zone for failover support, false to not have a standby replica.
 Default is true.",
          "type": "boolean",
          "default": true
        },
        "RDSOptionGroupName": {
          "description": "The option group that this DB instance is associated with.",
          "type": "string"
        },
        "RDSPerformanceInsights": {
          "type": "string",
```

```
          "description": "True to enable Performance Insights for the DB instance,
 false to not. Amazon RDS Performance Insights is a database performance tuning and
 monitoring feature that helps you assess the load on your database.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "RDSPerformanceInsightsKMSKey": {
          "type": "string",
          "description": "The Amazon resource name (ARN) of the KMS master key to
 use to encrypt Performance Insights data. Specify default to use the default RDS KMS
 Key.",
          "pattern": "^default$|^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]
{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
          "default": ""
        },
        "RDSPerformanceInsightsRetentionPeriod": {
          "type": "string",
          "description": "The amount of time, in days, to retain Performance Insights
 data. Valid values are 7 or 731 (2 years).",
          "enum": [
            "7",
            "731"
          ],
          "default": "7"
        },
        "RDSPreferredBackupWindow": {
          "description": "The daily time range during which automated backups are
 created if RDSBackupRetentionPeriod is set to a positive number. Must be in the format
 hh:mm-hh:mm (24-hour format), in Universal Coordinated Time (UTC). Must not conflict
 with the RDSPreferredMaintenanceWindow setting, and must be at least 30 minutes. If
 left blank a 30-minute window is selected at random from an 8-hour block of time for
 each AWS Region.",
          "type": "string",
          "pattern": "^$|^[0-9]{2}:[0-9]{2}-[0-9]{2}:[0-9]{2}$"
        },
        "RDSPort": {
          "description": "The port number on which the database accepts connections.
 Defaults vary per DB engine.",
          "type": "number"
        },
        "RDSPreferredMaintenanceWindow": {
```

```
          "description": "The weekly time range during which system maintenance can
 occur, in UTC. Must be in the format ddd:hh:mm-ddd:hh:mm (24-hour format). If left
 blank a 30-minute window selected at random from an 8-hour block of time for each AWS
 Region, occurring on a random day of the week.",
          "type": "string",
          "pattern": "^$|^[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$"
        },
        "RDSStorageEncrypted": {
          "description": "True to enable database encryption, false to not. Default is
 false.",
          "type": "boolean",
          "default": false
        },
        "RDSStorageEncryptionKey": {
          "description": "The ARN of the custom KMS key to encrypt the database if
 RDSStorageEncrypted = true. If RDSStorageEncrypted = true and you do not specify a
 RDSStorageEncryptionKey, RDS uses your default encryption key, which AWS KMS creates.
 Your AWS account has a different default encryption key for each AWS region.",
          "type": "string",
          "pattern": "^$|^arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/[a-f0-9]{8}-[a-f0-9]{4}-
[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$",
          "default": ""
        },
        "RDSStorageType": {
          "description": "Storage type for the RDS instance. If you specify io1, you
 must also include a value for the RDSIOPS parameter.",
          "type": "string",
          "enum": [
            "standard",
            "gp2",
            "io1",
            "gp3"
          ],
          "default": "gp2"
        },
        "RDSMaxAllocatedStorage": {
          "description": "The upper limit, in gibibytes (GiB), to which Amazon RDS can
 automatically scale the storage of the DB instance. This setting doesn't apply to RDS
 Custom. To learn more, see Amazon documentation on RDS DB instance storage.",
          "type": "string",
          "pattern": "2[0-9]|[3-9][0-9]|[1-9][0-9]{2,3}|[1-5][0-9]{4}|6[0-4][0-9]{3}|
65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-6]|^$"
        },
        "RDSSubnetIds": {
```

```
        "description": "Two or more subnet IDs for the RDS instance, in the form
subnet-0123abcd or subnet-01234567890abcdef, spanning at least two Availability
Zones.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
        },
        "minItems": 2,
        "maxItems": 20,
        "uniqueItems": true
      },
      "RDSTimezone": {
        "description": "The time zone of the DB instance. This is applicable only if
RDSDBEngine = sqlserver-ee, sqlserver-se, sqlserver-ex or sqlserver-web.",
        "type": "string"
      }
    },
    "additionalProperties": false,
    "metadata": {
      "ui:order": [
        "RDSDBEngine",
        "RDSLicenseModel",
        "RDSEngineVersion",
        "RDSInstanceType",
        "RDSTimezone",
        "RDSStorageType",
        "RDSAllocatedStorage",
        "RDSMaxAllocatedStorage",
        "RDSIOPS",
        "RDSDBInstanceIdentifier",
        "RDSDBName",
        "RDSMasterUsername",
        "RDSMasterUserPassword",
        "RDSMultiAZ",
        "RDSSubnetIds",
        "RDSPort",
        "RDSDBParameterGroupName",
        "RDSOptionGroupName",
        "RDSCharacterSetName",
        "RDSStorageEncrypted",
        "RDSStorageEncryptionKey",
        "RDSBackupRetentionPeriod",
        "RDSPreferredBackupWindow",
```

```
          "RDSAutoMinorVersionUpgrade",
          "RDSPerformanceInsights",
          "RDSPerformanceInsightsKMSKey",
          "RDSPerformanceInsightsRetentionPeriod",
          "RDSPreferredMaintenanceWindow",
          "RDSDeletionProtection"
        ]
      },
      "required": [
        "RDSAllocatedStorage",
        "RDSDBEngine",
        "RDSDBName",
        "RDSEngineVersion",
        "RDSMasterUsername",
        "RDSMasterUserPassword",
        "RDSSubnetIds"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "StackTemplateId",
    "Name",
    "TimeoutInMinutes",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2zebb2czoxpjd

Classifications:

- Management | Advanced stack components | Tag | Delete


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Resource Tags",
  "description": "Delete tags from existing, tagged resources: Autoscaling, EC2,
 Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally,
 CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS
 infrastructure stacks (stacks named mc-*) cannot have tags deleted with this change
 type.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateTags.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateTags"
      ],
      "default": "AWSManagedServices-UpdateTags"
    },
    "Region": {
      "description": "The AWS Region where the tagged resources are, in the form us-
east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ResourceArns": {
          "description": "A list of up to 50 Amazon resource names (ARNs), or the
 resource IDs, of the resources with tags to be deleted. Use resource ID only for these
 resource types: EC2 instance, EBS volume, EBS snapshot, AMI, and security group. Use
 the full ARN for all other supported resource types.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^(arn:aws:(autoscaling|ec2|elasticloadbalancing|logs|rds|s3|
redshift):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}):(|[0-9]{12}):.*)$|^(ami|i|vol|
sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "RemoveTags": {
          "description": "Up to fifty tag Keys to remove from the specified
 resource.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^((aws-migration-project-id)|(?![aA][mMwW][sS])[\\x00-\\x7F+]
{1,128})$",
            "minLength": 1,
            "maxLength": 127
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "ResourceArns",
          "RemoveTags"
        ]
      },
      "required": [
        "ResourceArns",
        "RemoveTags"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Parameters",
      "DocumentName"
    ]
  },
```

```
  "additionalProperties": false,
  "required": [
    "Region",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-2zqwr34epwzx1

Classifications:

- Deployment | Advanced stack components | RDS snapshot | Create (for cluster)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS DB cluster Snapshot",
  "description": "Create a snapshot of Amazon Aurora or Multi-AZ DB (Amazon RDS)
 cluster in available state. The snapshot will be encrypted with the same KMS key as
 the DB cluster, or unencrypted if the DB cluster is unencrypted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateDBClusterSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateDBClusterSnapshot"
      ],
      "default": "AWSManagedServices-CreateDBClusterSnapshot"
    },
    "Region": {
      "description": "The AWS Region in which the RDS DB cluster is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBClusterIdentifier": {
          "description": "The identifier for the RDS DB cluster that you are creating a
 snapshot of.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,62}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DBClusterSnapshotIdentifier": {
          "description": "A unique name for the RDS DB cluster snapshot.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!.*--)[a-zA-Z][a-zA-Z0-9-]{1,62}(?<!-)$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "DBClusterIdentifier",
          "DBClusterSnapshotIdentifier"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBClusterIdentifier",
        "DBClusterSnapshotIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
```

```
  ]
}
```

Schema for Change Type ct-2zxya20wmf5bf

Classifications:

- Management | Advanced stack components | KMS key | Delete (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete KMS key",
  "description": "Delete an AWS Key Management Service (KMS) Key from an AMS account.
 By default, there is a 30 day waiting period before the key is deleted; during that
 period, you can restore the key using the KMS Key Update change type.",
  "type": "object",
  "properties": {
    "KeyName": {
      "description": "The name of the KMS key to be deleted.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9:/_-]{1,256}$"
    },
    "Operation": {
      "description": "Must be Delete.",
      "type": "string",
      "default": "Delete",
      "enum": [
        "Delete"
      ]
    },
    "KeyDeletionWaitPeriod": {
      "description": "The waiting period, specified in number of days. After the
 waiting period ends, AWS KMS deletes the key from the account. Must be between 7 and
 30, inclusive. Default is 30.",
      "default": 30,
      "maximum": 30,
      "minimum": 7,
      "type": "integer"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
```

```
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "KeyName",
      "Operation",
      "KeyDeletionWaitPeriod",
      "Priority"
    ]
  },
  "required": [
    "KeyName",
    "Operation",
    "KeyDeletionWaitPeriod"
  ]
}
```

Schema for Change Type ct-3047c34zuvswh

Classifications:

- Management | Advanced stack components | Tag | Bulk update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Bulk Update Resource Tags",
  "description": "Bulk add tags to existing, supported resources: Autoscaling, EC2,
 Elastic Load Balancing, RDS and S3 buckets. AMS infrastructure stacks (stacks named
 mc-*) cannot have tags added with this change type. Use this with AWS Tag Editor when
 managing large numbers of tags (i.e. >50).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-BulkUpdateTags.",
      "type": "string",
```

```
      "enum": [
        "AWSManagedServices-BulkUpdateTags"
      ],
      "default": "AWSManagedServices-BulkUpdateTags"
    },
    "Region": {
      "description": "The AWS Region where the resources to be tagged are, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CsvS3Url": {
          "description": "The S3 bucket endpoint for the CSV file with the tag update
 details. The CSV file must be formatted to the correct format. Please see AMS tag
 documentation for the correct format of the CSV file.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^https?://[a-z0-9]([-.a-z0-9]+)[a-z0-9]\\.s3\\.([a-z]{2}-[a-
z]+-\\d{1}\\.)?amazonaws\\.com/[\\S]*",
            "minLength": 1,
            "maxLength": 5000
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "CsvS3Url"
        ]
      },
      "required": [
        "CsvS3Url"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
```

```
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-309eozh6lpkr8

Classifications:

- Deployment | Managed Firewall | Outbound (Palo Alto) | Create allow list

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Allow List",
  "description": "Create an allow list file for AMS managed Palo Alto firewall -
 Outbound.",
  "type": "object",
  "properties": {
    "RequestType": {
      "description": "Must be CreateAllowList.",
      "type": "string",
      "enum": [
        "CreateAllowList"
      ],
      "default": "CreateAllowList"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AllowListName": {
          "description": "A meaningful name for the allow list, cannot exceed 63
 characters.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9][a-zA-Z0-9-_]{0,62}$"
        }
```

```
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AllowListName"
        ]
      },
      "required": [
        "AllowListName"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "RequestType",
      "Parameters"
    ]
  },
  "required": [
    "RequestType",
    "Parameters"
  ]
}
```

Schema for Change Type ct-30bfiwxjku1nu

Classifications:

- Management | Advanced stack components | EBS snapshot | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete EBS Snapshots",
  "description": "Delete Elastic Block Store (EBS) snapshots. Because deleted snapshots
 cannot be restored, we recommend scheduling this RFC to provide a time period in which
 you could cancel the operation, if needed. At least one parameter must be specified.
 Note: If more than one parameter is used, only snapshots matching all used parameters
 are deleted. Snapshots created by AWS Backup service, used by AMIs, and snapshots
 created less than 60 days ago, cannot be deleted. If one or more snapshots cannot be
 deleted, execution fails. Up to 1000 snapshots can be deleted in one execution.",
  "type": "object",
```

```
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteEBSSnapshots.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteEBSSnapshots"
      ],
      "default": "AWSManagedServices-DeleteEBSSnapshots"
    },
    "Region": {
      "description": "The AWS Region to where the snapshots are, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Confirmation": {
      "description": "To confirm permanent deletion of the EBS snapshots, use delete
 permanently. If this parameter is unspecified, the RFC cannot be created.",
      "type": "string",
      "pattern": "^delete permanently$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SnapshotIds": {
          "description": "A list of up to 20 EBS snapshot IDs to delete, in
 the form snap-12345678 or snap-123456789012345ab. Use either this parameter or
 SnapshotIdCsvUrl, not both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$"
          },
          "minItems": 0,
          "maxItems": 20,
          "uniqueItems": true
        },
        "SnapshotIdCsvUrl": {
          "description": "A pre-signed S3 URL for the file with the list of snapshots
 to delete. The file must contain a comma separated list of up to 1000 snapshot IDs,
 in the form snap-12345678 or snap-123456789012345ab. Use either this parameter or
 SnapshotIds, not both.",
          "type": "array",
          "items": {
```

```
            "type": "string",
            "pattern": "^https?://[a-z0-9]([-.a-z0-9]+)[a-z0-9]\\.amazonaws\\.com/[\
\S]*$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "SnapshotCreationDate": {
          "description": "A snapshot creation date. Snapshots created before the
 specified date 00:00 UTC time are deleted. The date must be 60 or more days ago, in
 the form 2020-01-31.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^(20[12][0-9])-(0[1-9]|1[012])-(0[1-9]|[12][0-9]|3[01])$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "SnapshotTag": {
          "description": "A tag to filter snapshots for delete. The snapshots without
 the tag are not deleted. The tag is case sensitive and must be a single key-value
 pair, for example {\"Key\":\"Delete\",\"Value\":\"True\"}.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "\\{\"Key\":\"(?![aA][mMwW][sS])[a-zA-Z0-9\\s_.:/=+\\\\\\-@\\]*]
{1,127}\",\"Value\":\"[a-zA-Z0-9\\s_.:/=+\\\\\\-@\\]*]{1,127}\"\\}"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "SnapshotsWithoutVolumes": {
          "description": "True to delete only snapshots for which the source volumes no
 longer exist; False to delete all specified snapshots. Default is False.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "True",
              "False"
            ],
            "default": "False"
          },
```

```
          "minItems": 0,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SnapshotIds",
          "SnapshotIdCsvUrl",
          "SnapshotCreationDate",
          "SnapshotTag",
          "SnapshotsWithoutVolumes"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Confirmation",
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "Confirmation",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-30ecvfi3tq4k3

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create OpenID Connect provider

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create an OpenID Connect Provider",
  "description": "Create an IAM OpenID Connect provider for the Amazon Elastic
 Kubernetes Service (Amazon EKS) cluster.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleAssociateIAMOpenIDProvider-
Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleAssociateIAMOpenIDProvider-Admin"
      ],
      "default": "AWSManagedServices-HandleAssociateIAMOpenIDProvider-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterName": {
          "description": "The name of the Amazon EKS cluster to associate with the new
 OpenID Connect provider.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[A-Za-z0-9_-]{1,100}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ClusterName"
        ]
      },
      "required": [
        "ClusterName"
      ],
      "additionalProperties": false
```

```
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-30j78u6li9aqr

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Delete entity or policy (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete IAM Resource",
  "description": "Delete Identity and Access Management (IAM) users, roles or
 policies.",
  "type": "object",
  "properties": {
    "IAM Users": {
      "description": "A list of up to 10 IAM users to delete, in the ARN format.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^$|^arn:aws:iam::[0-9]{12}:user/([!-~]+/)*[\\w+=,.@-]+$",
        "minLength": 32,
        "maxLength": 607
      },
      "minItems": 0,
```

```
      "maxItems": 10
    },
    "IAM Roles": {
      "description": "A list of up to 10 IAM roles to delete, in the ARN format.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^$|^arn:aws:iam::[0-9]{12}:role/([!-~]+/)*[\\w+=,.@-]+$",
        "minLength": 32,
        "maxLength": 607
      },
      "minItems": 0,
      "maxItems": 10
    },
    "IAM Policies": {
      "description": "A list of up to 10 IAM policies to delete, in the ARN format.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^$|^arn:aws:iam::[0-9]{12}:policy/([!-~]+/)*[\\w+=,.@-]+$",
        "minLength": 34,
        "maxLength": 673
      },
      "minItems": 0,
      "maxItems": 10
    },
    "Operation": {
      "description": "Must be Delete.",
      "type": "string",
      "default": "Delete",
      "enum": [
        "Delete"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
```

```
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "IAM Users",
      "IAM Roles",
      "IAM Policies",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "Operation"
  ]
}
```

Schema for Change Type ct-31eb7rrxb7qju

Classifications:

- Management | Advanced stack components | S3 storage | Add replication rule

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Replication Rule",
  "description": "Add an S3 replication rule to the specified S3 bucket.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-PutReplicationRule.",
      "type": "string",
      "enum": [
        "AWSManagedServices-PutReplicationRule"
      ],
      "default": "AWSManagedServices-PutReplicationRule"
    },
    "Region": {
      "description": "The AWS Region in which the source bucket is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
```

```
    "Parameters": {
      "type": "object",
      "properties": {
        "ReplicationRuleName": {
          "description": "The replication rule name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[A-Za-z0-9_-]+$"
          },
          "maxItems": 1
        },
        "DestinationAccount": {
          "description": "The destination S3 bucket account ID, use the same account ID
  if the destination bucket is within the current account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "maxItems": 1
        },
        "DestinationBucketName": {
          "description": "The destination S3 bucket name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z0-9]$",
            "minLength": 3,
            "maxLength": 63
          },
          "maxItems": 1
        },
        "SourceBucketName": {
          "description": "The source S3 bucket name.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z0-9]([-.a-z0-9]+)[a-z0-9]$",
            "minLength": 3,
            "maxLength": 63
          },
          "maxItems": 1
        },
```

```
        "ReplicationRole": {
          "description": "The ARN of the role that allows S3 to perform the replication
 on your behalf.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:role/[A-Za-z0-9_-]+
$"
          },
          "maxItems": 1
        },
        "DecryptObjectKMSKey": {
          "description": "The KMS key(s) used to decrypt objects in the source S3
 bucket.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
            "default": ""
          }
        },
        "EncryptReplicaKMSKey": {
          "description": "The KMS key used to encrypt destination objects.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$",
            "default": ""
          },
          "maxItems": 1
        },
        "OwnerTranslation": {
          "description": "True to change replica ownership to the AWS account that owns
 the destination bucket, false to not change replica ownership. This parameter cannot
 be left blank.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ],
```

```
            "default": "false"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Prefix": {
          "description": "An object key name prefix that identifies the subset of
objects to which the rule applies; for example, 'customer-'.",
          "type": "array",
          "items": {
            "type": "string",
            "default": ""
          },
          "maxItems": 1
        },
        "Priority": {
          "description": "S3 uses the rule priority to determine which rule to apply.
The higher the number, the higher the priority. Default rule priority is 1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([1-9]|[1-9][0-9]{1,})$",
            "default": "1"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ReplicationRuleName",
          "SourceBucketName",
          "DestinationAccount",
          "DestinationBucketName",
          "ReplicationRole",
          "OwnerTranslation",
          "DecryptObjectKMSKey",
          "EncryptReplicaKMSKey",
          "Prefix",
          "Priority"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ReplicationRuleName",
```

```
        "SourceBucketName",
        "DestinationAccount",
        "DestinationBucketName",
        "ReplicationRole"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-31eyj2hlvqjwu

Classifications:

- Management | Advanced stack components | RDS database stack | Update Performance Insights (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Performance Insights.",
  "description": "Update Performance Insights for a DB instance or Multi-AZ DB cluster.
 Amazon RDS Performance Insights is a database performance tuning and monitoring
 feature that helps you assess the load on your database. You can change settings,
 enable, or disable the feature.",
  "type": "object",
  "properties": {
    "DBIdentifierArn": {
      "description": "The Amazon Resource Name (ARN) that uniquely identifies the DB
 instance or cluster.",
```

```
      "type": "string",
      "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:([a-z]{2}((-gov))?-[a-z]+-\\d{1}):
[0-9]{12}:(db|cluster):[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
    },
    "PerformanceInsights": {
      "type": "string",
      "description": "True to enable Performance Insights for the DB instance, false to
 not. Enabling Performance Insights doesn't cause downtime, a reboot, or a failover.",
      "enum": [
        "true",
        "false"
      ]
    },
    "PerformanceInsightsKMSKeyId": {
      "type": "string",
      "description": "The Amazon resource name (ARN) of the KMS master key to use to
 encrypt Performance Insights data. Specify default to use the default RDS KMS Key.",
      "pattern": "^default$|^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z0-9-]+:[0-9]
{12}:key/)?[a-f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$",
      "default": "default"
    },
    "PerformanceInsightsRetentionPeriod": {
      "type": "string",
      "description": "The number of days to retain Performance Insights data. The
 default is 7 days",
      "enum": [
        "7 days",
        "1 month",
        "2 months",
        "3 months",
        "4 months",
        "5 months",
        "6 months",
        "7 months",
        "8 months",
        "9 months",
        "10 months",
        "11 months",
        "12 months",
        "13 months",
        "14 months",
        "15 months",
        "16 months",
        "17 months",
```

```
        "18 months",
        "19 months",
        "20 months",
        "21 months",
        "22 months",
        "23 months"
      ],
      "default": "7 days"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DBIdentifierArn",
      "PerformanceInsights",
      "PerformanceInsightsKMSKeyId",
      "PerformanceInsightsRetentionPeriod",
      "Priority"
    ]
  },
  "required": [
    "DBIdentifierArn",
    "PerformanceInsights"
  ]
}
```

Schema for Change Type ct-33ste5yc7hprs

Classifications:

- Deployment | Managed landing zone | Management account | Create custom SCP (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Custom SCP",
  "description": "Create a custom service control policy (SCP) to manage permissions
 across AWS organization.",
  "type": "object",
  "properties": {
    "TargetId": {
      "description": "The unique identifier (ID) of the root, or organizational unit
 (OU), or account, that you want to attach the SCP to. For information on creating a
 SCP, refer to AWS documentation.",
      "type": "string",
      "pattern": "^ou-[0-9a-z]{4,32}-[a-z0-9]{8,32}$|^r-[0-9a-z]{4,32}$|^[0-9]{12}$"
    },
    "CustomServiceControlPolicy": {
      "description": "The JSON contents of the SCP that you want to attach to the
 target.",
      "type": "string",
      "maxLength": 5000
    },
    "SCPDescription": {
      "description": "A description of the SCP to be attached to the provided
 target.",
      "type": "string"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "SCPDescription",
      "TargetId",
      "CustomServiceControlPolicy",
      "Priority"
    ]
```

```
  },
  "additionalProperties": false,
  "required": [
    "TargetId",
    "CustomServiceControlPolicy"
  ]
}
```

Schema for Change Type ct-34alumbtv2b9p

Classifications:

- Management | Advanced stack components | Stack patching configuration | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update stack patching configuration",
  "description": "Use to update patch configuration.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "HealthyHostThreshold": {
      "exclusiveMaximum": true,
      "description": "The minimum health threshold, in decimal, of available instances
 within a stack that must be maintained during patching.",
      "maximum": 1,
      "type": "number",
      "minimum": 0
    },
    "MaintenanceWindow": {
      "description": "The monthly maintenance window within which patching will occur,
 in UTC.",
      "type": "object",
      "properties": {
        "DayOfWeek": {
          "description": "Day of the week (1 to 7 == Monday to Sunday).",
          "maximum": 7,
          "type": "integer",
          "minimum": 1
        },
        "DurationInMinutes": {
          "description": "Duration of the window in minutes.",
```

```
          "maximum": 1440,
          "type": "integer",
          "minimum": 60
        },
        "Minute": {
          "description": "Minute of the hour of the day that the window will begin.",
          "maximum": 59,
          "type": "integer",
          "minimum": 0
        },
        "Hour": {
          "description": "Hour of the day that the window will begin.",
          "maximum": 23,
          "type": "integer",
          "minimum": 0
        },
        "WeekOfMonth": {
          "description": "Week of the month that the window will reside within (1 ==
 first week of the month, 4 == 4th week of the month).",
          "maximum": 4,
          "type": "integer",
          "minimum": 1
        }
      }
    },
    "StackId": {
      "pattern": "^stack-[a-zA-Z0-9]{17}$",
      "description": "The ID of the stack to perform the task on, in the form of
 stack-12345678901234567.",
      "type": "string"
    }
  },
  "required": [
    "StackId"
  ]
}
```

Schema for Change Type ct-34jldf2qihaic

Classifications:

- Management | Advanced stack components | EBS Volume | Attach

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Attach EBS Volume",
  "description": "Attach an EBS volume to an EC2 instance. This change type provides an
 option that attempts to remediate drift in the CloudFormation stack where the volume
 is being attached, but that option, RemediateStackDrift, does not work on volumes
 created using the CloudFormation ingest change type (ct-36cn2avfrrj9v).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AttachEBSVolume.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AttachEBSVolume"
      ],
      "default": "AWSManagedServices-AttachEBSVolume"
    },
    "Region": {
      "description": "The AWS Region where the EBS Volume is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance, in the form
 i-1234567890abcdef0.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8,17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "VolumeId": {
          "description": "The ID of the EBS volume, in the form
 vol-1234567890abcdef0.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^vol-([0-9a-f]{8}|[0-9a-f]{17})$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DeviceName": {
          "description": "The device name where the volume is to be attached, e.g. /
dev/sdf or xvdg. If no device name is included, one is chosen for you.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(/dev/sd[a-z][1-15]{0,1})|xvd[a-z]$|/dev/xvd[a-z]$|^$"
          },
          "minItems": 0,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused
 by volume attachment. False to not attempt drift remediation. Drift remediation can
 be performed only on CloudFormation stacks that were created using a CT other than
 the Ingestion CT ct-36cn2avfrrj9v and that are in sync with the definitions in the
 stack template prior to the volume attachment. Set to False to attach a volume in an
 ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "VolumeId",
          "InstanceId",
          "DeviceName",
          "RemediateStackDrift"
        ]
      },
```

```
      "required": [
        "VolumeId",
        "InstanceId",
        "RemediateStackDrift"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-34sxfo53yuzah

Classifications:

- Management | Custom Stack | Stack from CloudFormation Template | Remediate drift (review required)
- Management | Standard stacks | Stack | Remediate drift (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remediate Stack Drift",
  "description": "Remediate the drift (out-of-band changes) in a stack, bringing the
 stack in sync and enabling you to perform future updates using the available Update
 CTs. Drift remediation can be performed on EC2 resource types.",
  "type": "object",
  "properties": {
    "StackName": {
```

```
      "description": "The name of the stack to remediate the drift, in the form of
 stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{8}$|^stack-[a-z0-9]{17}$"
    },
    "DryRun": {
      "description": "True to perform drift remediation in dry run mode, false to
 perform drift remediation not in dry run mode. Default is false. Dry run mode checks
 if the stack drift can be remediated or not, but does not attempt remediation. Note
 that, when DryRun=true, reserved stack outputs for drift remediation, in the form of
 AMSCFNDriftRemediationBuildReferences95556500d5, can be added or updated. To learn
 more about outputs, see AWS CloudFormation documentation.",
      "type": "boolean",
      "default": false
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "StackName",
      "DryRun",
      "Priority"
    ]
  },
  "required": [
    "StackName"
  ]
}
```

Schema for Change Type ct-35p977vul06df

Classifications:

- Management | Advanced stack components | Network Load Balancer | Add listener certificate


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add NLB Listener Certificate",
  "description": "Add a certificate to the specified Network Load Balancer (NLB)
 listener. Use the RemediateStackDrift parameter for the automation to try to remediate
 drift, if it is introduced.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddCertificateToElbv2Listener.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddCertificateToElbv2Listener"
      ],
      "default": "AWSManagedServices-AddCertificateToElbv2Listener"
    },
    "Region": {
      "description": "The AWS Region where the network load balancer listener is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ListenerArn": {
          "description": "The Amazon Resource Name (ARN) of the listener in
 the form arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/net/
sample/1234567890abcdfe/1234567890abcdfe.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):elasticloadbalancing:[a-z]{2}-[a-
z]+-[0-9]{1}:[0-9]{12}:listener/net/[A-Za-z0-9-]+/[a-z0-9-]+/[a-z0-9-]+$"
          },
          "minItems": 1,
```

```
          "maxItems": 1
        },
        "CertificateArn": {
          "description": "The Amazon Resource Name (ARN) of the certificate in the form
 arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IsDefault": {
          "description": "True to set the certificate as the default certificate on the
 listener, False to not set the certificate as the default certificate on the listener.
 Default value is False.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused
 by adding the certificate to the Load Balancer listener. False to not attempt drift
 remediation. Drift remediation can be performed only on CloudFormation stacks that
 were created using a CT other than the Ingestion CT ct-36cn2avfrrj9v and that are
 in sync with the definitions in the stack template prior to adding certificate to
 the Load Balancer listener. Set to False to add the certificate to the Load Balancer
 listener in an ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
```

```
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ListenerArn",
          "CertificateArn",
          "IsDefault",
          "RemediateStackDrift"
        ]
      },
      "additionalProperties": false,
      "required": [
        "CertificateArn",
        "ListenerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-361tlo1k7339x

Classifications:

- Management | Custom Stack | Stack from CloudFormation Template | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update CloudFormation Stack",
  "description": "Update the template and/or parameters of a CFN stack. To only update
 the parameters in an existing stack a modified CFN template is not required, modified
 parameters can be provided instead. Values for existing parameters are overwritten,
 values for new parameters are added. To add, delete or modify a resource, or to change
 attributes not referenced through a parameter, use a modified CFN template. If the
 update would result in a resource in the stack being replaced or removed, the RFC
 fails and requires approval through the \"Approve ChangeSet and update CloudFormation
 stack\" CT (ct-1404e21baa2ox).",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "Identifier of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "Identifier for the existing CloudFormation-based stack to be
 updated.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "CloudFormationTemplateS3Endpoint": {
      "description": "The Amazon S3 bucket URL for the CloudFormation template you
 want to deploy. The template must be accessible from this account or provided as
 a pre-signed Amazon S3 URL. To update the template for an existing stack, provide
 either an Amazon S3 URL for the template in this option, or an inline template in the
 CloudFormationTemplate option.",
      "type": "string",
      "minLength": 1,
      "pattern": "^[\\s]*https?://[\\S]*[\\s]*$|^[\\s]*$",
      "maxLength": 2047
    },
    "CloudFormationTemplate": {
      "description": "The CloudFormation template that you have configured
 to create or update the resources that you want. To update the template for
 an existing stack, provide either an Amazon S3 URL for the template in the
 CloudFormationTemplateS3Endpoint option, or an inline template in this option.",
      "type": "string",
      "minLength": 1,
```

```
      "pattern": "^(?![\\s]*https?)[\\S\\s]*$",
      "maxLength": 20000
    },
    "TemplateParameters": {
      "description": "Parameters (key/value pairs) from the CloudFormation template
used to configure the stack. Unspecified parameters retain their current values. New
parameters defined in the updated template must either have a default value or a value
provided here.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string"
          },
          "Value": {
            "type": "string"
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "uniqueItems": true
    },
    "AutoApproveRiskyUpdates": {
      "description": "Logical IDs in your template that represent resources for which a
high-risk update should be automatically approved, without requiring your approval of
a change set. High-risk is defined as an update that could cause resource deletion or
replacement. If the stack update includes high-risk changes that are not included in
this list, you will be required to approve a change set to execute the change through
the \"Approve ChangeSet and update CloudFormation stack\" CT (ct-1404e21baa2ox).",
      "type": "array",
      "items": {
        "type": "string"
      },
```

```
      "uniqueItems": true
    },
    "BypassDriftCheck": {
      "description": "Logical IDs in your template that represent drifted, or drift
 unsupported resources for which the drift check should be bypassed before updating
 the resource. If the stack update includes updating drifted, or drift unsupported
 resources that are not included in this list, the update will fail. Carefully inspect
 the drift report before bypassing the drift check for the resources to be updated.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
 the change. This does not prolong execution, but the RFC fails if the change is not
 completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 1080,
      "default": 360
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "CloudFormationTemplateS3Endpoint",
      "CloudFormationTemplate",
      "TemplateParameters",
      "AutoApproveRiskyUpdates",
      "TimeoutInMinutes",
      "BypassDriftCheck"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-361vpyun9a9dd

Classifications:

- Deployment | Monitoring and notification | CloudWatch | Create alarms


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create CloudWatch alarms",
  "description": "Create one or more CloudWatch alarms. For detailed information on
 CloudWatch alarm properties, see AWS documentation \"Creating CloudWatch Alarms\".",
  "type": "object",
  "properties": {
    "Alarms": {
      "description": "Parameters for one or more CloudWatch alarms.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "ActionsEnabled": {
            "description": "True for specified CloudWatch supported actions, including
 SNS topic actions, to be triggered. False for actions to not be triggered. For AMS
 to monitor your alarms, the SNS topic must be added to each configured action and
 ActionsEnabled must be set to true. To request that AMS perform actions not supported
 by CloudWatch, provide AMS with detailed instructions on handling the alarm in a
 service request after the alarm is created.",
            "type": "boolean"
          },
          "AlarmActions": {
            "description": "The Amazon Resource Name (ARN) of existing actions to
 execute when this alarm transitions to the ALARM state from any other state. If
 unspecified, no action is taken when this alarm transitions to the ALARM state. For
 AWS Managed Services (AMS) to monitor the alarms, include your AMS MMS SNS topic, in
 the form [\"arn:aws:sns:${REGION}:${ACCOUNT_ID}:MMS-Topic\"]",
            "type": "array",
            "items": {
              "type": "string",
              "minLength": 1,
              "maxLength": 1024
            },
            "maxItems": 5,
            "uniqueItems": true
```

```
        },
        "AlarmDescription": {
          "description": "A meaningful description for the alarm.",
          "type": "string",
          "minLength": 0,
          "maxLength": 1024
        },
        "AlarmName": {
          "description": "A name for the alarm. The name must be unique within the
AWS account.",
          "type": "string",
          "minLength": 1,
          "maxLength": 255
        },
        "ComparisonOperator": {
          "description": "The operation to use when comparing the Statistic
and Threshold values that you specify. The specified Statistic value is used as
the first operand. Options: GreaterThanOrEqualToThreshold, GreaterThanThreshold,
LessThanThreshold, LessThanOrEqualToThreshold.",
          "type": "string",
          "enum": [
            "GreaterThanOrEqualToThreshold",
            "GreaterThanThreshold",
            "LessThanThreshold",
            "LessThanOrEqualToThreshold"
          ]
        },
        "DatapointsToAlarm": {
          "description": "The number of datapoints that must be breaching to
trigger the alarm. Required if you are setting an \"M out of N\" alarm. If you set
DatapointsToAlarm and EvaluationPeriod as different values, you are setting an \"M out
of N\" alarm (DatapointsToAlarm is \"M\", EvaluationPeriod is \"N\").",
          "type": "integer",
          "minimum": 1
        },
        "Dimensions": {
          "description": "The dimensions (arbitrary name/value pairs) for the metric
associated with the alarm.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "Name": {
                "description": "The name of the dimension.",
```

```
                "type": "string",
                "minLength": 1,
                "maxLength": 255
              },
              "Value": {
                "description": "The value representing the dimension measurement.",
                "type": "string",
                "minLength": 1,
                "maxLength": 255
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "Name",
                "Value"
              ]
            },
            "required": [
              "Name",
              "Value"
            ]
          },
          "maxItems": 10,
          "uniqueItems": true
        },
        "EvaluateLowSampleCountPercentile": {
          "description": "For alarms based on percentiles. Options: evaluate, ignore.
For the alarm state to not change during periods with too few data points to be
statistically significant, set ignore. For the alarm to always be evaluated, and
possibly change state, no matter how many data points are available, set evaluate or
leave blank.",
          "type": "string",
          "minLength": 1,
          "maxLength": 255,
          "enum": [
            "evaluate",
            "ignore"
          ]
        },
        "EvaluationPeriods": {
          "description": "The number of consecutive data points that must be breached
to trigger the alarm. For an \"M out of N\" alarm, this value is the N.",
          "type": "integer",
```

```
            "minimum": 1
          },
          "ExtendedStatistic": {
            "description": "For alarms based on percentiles. The percentile statistic
 for the metric associated with the alarm. Specify a value between p0.0 and p100. You
 must specify either this, or Statistic, but not both.",
            "type": "string",
            "pattern": "p(\\d{1,2}(\\.\\d{0,2})?|100)"
          },
          "InsufficientDataActions": {
            "description": "The Amazon Resource Name (ARN) of one or more
 existing CloudWatch alarm actions to execute when this alarm transitions to the
 INSUFFICIENT_DATA state from any other state. If unspecified, no action is taken when
 this alarm transitions to the INSUFFICIENT_DATA state. For AWS Managed Services (AMS)
 to monitor the alarm, include your AMS MMS SNS topic, in the form [\"arn:aws:sns:
${REGION}:${ACCOUNT_ID}:MMS-Topic\"]",
            "type": "array",
            "items": {
              "type": "string",
              "minLength": 1,
              "maxLength": 1024
            },
            "maxItems": 5,
            "uniqueItems": true
          },
          "MetricName": {
            "description": "An existing standard or custom CloudWatch metric for the
 alarm to track. For a list of AWS CloudWatch metrics, see AWS CloudWatch metrics
 documentation. To use a custom CloudWatch metric, see your CloudWatch console.",
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          },
          "Namespace": {
            "description": "An existing standard or custom CloudWatch namespace for the
 alarm. For a list of AWS namespaces, see AWS documentation. To use a custom namespace,
 see your CloudWatch console Metrics area.",
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          },
          "OkActions": {
            "description": "The Amazon Resource Name (ARN) of one or more existing
 CloudWatch alarm actions to execute when this alarm transitions to the OK state from
```

```
any other state. If unspecified, no action is taken when this alarm transitions to the
OK state. For AWS Managed Services (AMS) to monitor the alarm, include your AMS MMS
SNS topic, in the form [\"arn:aws:sns:${REGION}:${ACCOUNT_ID}:MMS-Topic\"]",
          "type": "array",
          "items": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1024
          },
          "maxItems": 5,
          "uniqueItems": true
        },
        "Period": {
          "description": "The period, in seconds, over which the specified statistic
is applied. Valid values are 10, 30, and any multiple of 60. Be sure to specify 10 or
30 only for metrics that are stored by a PutMetricData call with a StorageResolution
of 1.",
          "type": "integer",
          "minimum": 1
        },
        "Statistic": {
          "description": "The statistic for the metric associated with the alarm;
does not apply to percentile metrics. Options: SampleCount, Average, Sum, Minimum,
Maximum. For percentile statistics, use parameter ExtendedStatistic. You must specify
either this property, or ExtendedStatistic, but not both.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ]
        },
        "Threshold": {
          "description": "The value against which the specified statistic is
compared.",
          "type": "number"
        },
        "TreatMissingData": {
          "description": "How this alarm handles missing data points. Options:
breaching, notBreaching, ignore, missing. If unspecified, the default behavior,
missing, is used.",
          "type": "string",
```

```
            "enum": [
              "breaching",
              "notBreaching",
              "ignore",
              "missing"
            ],
            "minLength": 1,
            "maxLength": 255
          },
          "Unit": {
            "description": "The unit of measure for the statistic. Valid options are
 provided in the AWS Java SDK page for Enum StandardUnit.",
            "type": "string",
            "enum": [
              "Seconds",
              "Microseconds",
              "Milliseconds",
              "Bytes",
              "Kilobytes",
              "Megabytes",
              "Gigabytes",
              "Terabytes",
              "Bits",
              "Kilobits",
              "Megabits",
              "Gigabits",
              "Terabits",
              "Percent",
              "Count",
              "Bytes/Second",
              "Kilobytes/Second",
              "Megabytes/Second",
              "Gigabytes/Second",
              "Terabytes/Second",
              "Bits/Second",
              "Kilobits/Second",
              "Megabits/Second",
              "Gigabits/Second",
              "Terabits/Second",
              "Count/Second",
              "None"
            ]
          }
        },
```

```
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "AlarmName",
            "AlarmDescription",
            "Namespace",
            "MetricName",
            "EvaluationPeriods",
            "Period",
            "ComparisonOperator",
            "Threshold",
            "Statistic",
            "Dimensions",
            "Unit",
            "DatapointsToAlarm",
            "EvaluateLowSampleCountPercentile",
            "ExtendedStatistic",
            "TreatMissingData",
            "ActionsEnabled",
            "AlarmActions",
            "InsufficientDataActions",
            "OkActions"
          ]
        },
        "required": [
          "AlarmName",
          "ComparisonOperator",
          "EvaluationPeriods",
          "MetricName",
          "Namespace",
          "Period",
          "Threshold"
        ]
      }
    },
    "Region": {
      "description": "The AWS Region to create the alarm or set of alarms in.",
      "type": "string"
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Region",
```

```
      "Alarms"
    ]
  },
  "required": [
    "Alarms",
    "Region"
  ]
}
```

Schema for Change Type ct-369odosk0pd9w

Classifications:

- Management | Directory Service | Directory | Share directory

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Share Directory",
  "description": "Share a specified directory in your AWS account (directory owner)
 with another AWS account (directory consumer). Run this in your Shared Service account
 that has Managed Active Directory. This change type is only supported for multi-
account landing zone (MALZ).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-ShareDirectory.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ShareDirectory"
      ],
      "default": "AWSManagedServices-ShareDirectory"
    },
    "Region": {
      "description": "The AWS Region where the directory is located, in the form of us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DirectoryId": {
```

```
          "description": "Identifier of the AWS Managed Microsoft Active directory that
 you want to share with another AWS account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^d-[0-9a-f]{10}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "TargetAccountId": {
          "description": "Identifier for the directory consumer account to share the
 directory with.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "DirectoryId",
          "TargetAccountId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DirectoryId",
        "TargetAccountId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
```

```
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-36cn2avfrrj9v

Classifications:

- Deployment | Ingestion | Stack from CloudFormation Template | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Stack From CloudFormation (CFN) Template",
  "description": "Create a stack by pointing to a customized CloudFormation (CFN)
 template in an S3 bucket, or by pasting the contents of that template as input to this
 change type.",
  "type": "object",
  "properties": {
    "CloudFormationTemplate": {
      "description": "Your customized CFN template, copied directly into
 this input parameter. Use this parameter, CloudFormationTemplate, or the
 CloudFormationTemplateS3Endpoint parameter. Do not use both.",
      "type": "string",
      "minLength": 1,
      "pattern": "^(?![\\s]*https?)[\\S\\s]*$",
      "maxLength": 20000
    },
    "CloudFormationTemplateS3Endpoint": {
      "description": "The S3 bucket endpoint for the CloudFormation template you want
 to use. The bucket must be in the same account that you are using, or have a presigned
 URL.",
      "type": "string",
      "minLength": 1,
      "pattern": "^[\\s]*https?://[\\S]*[\\s]*$|^[\\s]*$",
      "maxLength": 2047
    },
    "Parameters": {
      "description": "Add up to sixty parameters (parameter name/value pairs) to
 supply alternate values for parameters in your customized CloudFormation template.
 By providing the parameters this way, you can reuse your CloudFormation template with
```

```
different parameter values when needed and can update any parameter value with the CFN
Stack Update change type.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Name": {
           "type": "string"
         },
         "Value": {
           "type": "string"
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Name",
           "Value"
         ]
       },
       "required": [
         "Name",
         "Value"
       ]
     },
     "minItems": 0,
     "maxItems": 60,
     "uniqueItems": true
   },
   "Description": {
     "description": "Meaningful information about the stack to be created.",
     "type": "string",
     "minLength": 1,
     "maxLength": 500
   },
   "Name": {
     "description": "A name for the stack; this becomes the Stack Name in the AMS
console.",
     "type": "string",
     "minLength": 1,
     "maxLength": 255
   },
   "VpcId": {
```

```
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the stack.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
```

```
      "maximum": 360,
      "default": 360
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "CloudFormationTemplateS3Endpoint",
      "CloudFormationTemplate",
      "Parameters",
      "TimeoutInMinutes",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "Name",
    "VpcId",
    "TimeoutInMinutes"
  ]
}
```

Schema for Change Type ct-36emj2uapfbu8

Classifications:

- Deployment | Standalone resources | EC2 instance | Create for WIGS (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EC2 for WIGS",
  "description": "Create an Amazon Elastic Compute Cloud (EC2) instance for use with
 Workload Ingest (WIGS) change type (ct-257p9zjk14ija). For information, see AMS
 documentation on WIGS in the AMS Application Developer's Guide.",
  "type": "object",
  "properties": {
    "InstanceVpcId": {
      "description": "The ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
```

```
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "InstanceNameTagValue": {
      "description": "A value for the Instance Name Tag Key.",
      "type": "string",
      "pattern": "^(?!([aA][mMwW][sS]|mc-))[a-zA-Z0-9_.@-]{0,256}$"
    },
    "InstanceAmiId": {
      "description": "The AMI to use to create the EC2 instance, in the form
ami-0123abcd or ami-01234567890abcdef.",
      "type": "string",
      "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$"
    },
    "InstanceEBSOptimized": {
      "description": "True for the instance to be optimized for Amazon Elastic Block
Store I/O, false for it to not be. If you set this to true, choose an InstanceType
that supports EBS optimization.",
      "type": "boolean",
      "default": false
    },
    "InstanceType": {
      "description": "The type of EC2 instance to deploy. If InstanceEBSOptimized =
true, specify an InstanceType that supports EBS optimization.",
      "type": "string",
      "pattern": "^[a-z0-9]+\\.[a-z0-9]+$",
      "default": "t3.large"
    },
    "InstanceRootVolumeSize": {
      "description": "The size of the root volume for the instance. Defaults to 20 GiB
for Linux, and 60 GiB for Windows.",
      "type": "number",
      "minimum": 20,
      "maximum": 16000
    },
    "InstanceSubnetId": {
      "description": "The subnet that you want to launch the instance into, in the form
subnet-0123abcd or subnet-01234567890abcdef.",
      "type": "string",
      "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
```

```
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "InstanceVpcId",
      "InstanceSubnetId",
      "InstanceAmiId",
      "InstanceType",
      "InstanceEBSOptimized",
      "InstanceRootVolumeSize",
      "InstanceNameTagValue",
      "Priority"
    ]
  },
  "required": [
    "InstanceVpcId",
    "InstanceAmiId",
    "InstanceSubnetId"
  ]
}
```

Schema for Change Type ct-36jq7gvwyty8h

Classifications:

- Management | Advanced stack components | RDS database stack | Update MultiAZ setting

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change RDS MultiAZ Setting",
  "description": "Change the DB instance MultiAZ value through direct API calls.
 The MultiAZ setting determines whether or not the DB instance is deployed across
 multiple availability zones (AZs). The RDS instance can be standalone or belong to a
 CloudFormation stack; in the latter case, the change might cause stack drift. To avoid
```

```
 causing stack drift, please use ct-12w49boaiwtzp instead, or ct-361tlo1k7339x if the
 RDS instance was provisioned via CFN ingestion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateRDSMultiAZ.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateRDSMultiAZ"
      ],
      "default": "AWSManagedServices-UpdateRDSMultiAZ"
    },
    "Region": {
      "description": "The AWS Region in which the resource is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBInstanceIdentifier": {
          "description": "The identifier of the RDS database instance; for example,
 mydbinstance.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(mc|ams|awsms)-)[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]
{0,62}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "MultiAZ": {
          "description": "True for the DB instance to be deployed across multiple AZs,
 false for it to not.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        },
        "ApplyImmediately": {
```

```
          "description": "True to apply the change immediately, false to schedule the
 change for the next maintenance window.",
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "DBInstanceIdentifier",
          "MultiAZ",
          "ApplyImmediately"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBInstanceIdentifier",
        "MultiAZ",
        "ApplyImmediately"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-36x3u7v2oklwd

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create account alias


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create AWS Account Alias",
  "description": "Create an AWS account alias. Note that an AWS account can have only
 one alias. This operation fails if the AWS account already has an alias. To update
 an existing account alias, use the Update Account Alias (ct-3skaisgnq0pf8) change
 type.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateAccountAlias.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateAccountAlias"
      ],
      "default": "AWSManagedServices-CreateAccountAlias"
    },
    "Region": {
      "description": "The AWS Region where the account is, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AWSAccountAlias": {
          "description": "The alias name for the AWS account to create.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?=[a-zA-Z0-9-]{3,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
```

```
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AWSAccountAlias"
        ]
      },
      "required": [
        "AWSAccountAlias"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-36zubwzxp44a4

Classifications:

- Management | Advanced stack components | Bastions | Add CIDR ingress (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add CIDR Ingress",
  "description": "Add RDP or SSH bastion ingress Classless Inter-Domain Routing (CIDR)
 allow lists.",
  "type": "object",
  "properties": {
    "BastionType": {
      "description": "The bastion type to update.",
```

```
      "type": "string",
      "enum": [
        "RDP Bastion",
        "SSH Bastion"
      ]
    },
    "IngressCIDRAddresses": {
      "description": "The CIDR ingress IP addresses to be allowed.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
      },
      "minItems": 1,
      "maxItems": 3
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "BastionType",
      "IngressCIDRAddresses",
      "Priority"
    ]
  },
  "additionalProperties": false,
  "required": [
    "BastionType",
    "IngressCIDRAddresses"
  ]
}
```

Schema for Change Type ct-379uwo67vbvng

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Update SAML identity provider


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update SAML Identity Provider",
  "description": "Update IAM identity provider using the SAML metadata document file
 that you stored in your chosen S3 bucket.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleUpdateSamlProvider-Admin",
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleUpdateSamlProvider-Admin"
      ],
      "default": "AWSManagedServices-HandleUpdateSamlProvider-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SAMLMetadataDocumentURL": {
          "description": "The S3 URL of the SAML metadata document file, in the form
 s3://bucketname/path/to/saml-metadata.xml.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^s3://[a-z0-9]([-.a-z0-9]+)[a-z0-9]/.+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SAMLProviderArn": {
```

```
          "description": "The ARN of the SAML provider.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):iam::[0-9]{12}:saml-provider/[\
\w._-]{1,128}"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SAMLProviderBackup": {
          "description": "True for a backup of the SAML provider metadata to be taken
 before deleting, False for no backup to be taken. Default is True.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SAMLMetadataDocumentURL",
          "SAMLProviderArn",
          "SAMLProviderBackup"
        ]
      },
      "required": [
        "SAMLMetadataDocumentURL",
        "SAMLProviderArn"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
```

```
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-37bq2l9c8fzxv

Classifications:

- Management | Advanced stack components | Target group | Detach instances

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Detach Instance From Target Group",
  "description": "Detach an instance, or instances, from the specified port of a target
 group (ALB or NLB).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-DetachInstancesFromTargetGroup",
      "type": "string",
      "enum": [
        "AWSManagedServices-DetachInstancesFromTargetGroup"
      ],
      "default": "AWSManagedServices-DetachInstancesFromTargetGroup"
    },
    "Region": {
      "description": "The AWS Region where the target group and instances are located,
 in the form of us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstancesIds": {
```

```
          "description": "The instance, or instances', IDs (up to 20) to be detached
 from the required target group, in the form of i-1234abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8,17}$"
          },
          "maxItems": 20
        },
        "InstancesPort": {
          "description": "The port to detach the instance, or instances, from.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4][0-9]
{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "maxItems": 1
        },
        "TargetGroupArn": {
          "description": "The target group Amazon Resource Name (ARN), in the
 form of arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:elasticloadbalancing:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstancesIds",
          "InstancesPort",
          "TargetGroupArn"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstancesIds",
        "InstancesPort",
        "TargetGroupArn"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-37kcp2v1mriu6

Classifications:

- Management | Advanced stack components | EC2 instance stack | Replace instance profile

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Replace Instance Profile",
  "description": "Replace the instance profile of an EC2 instance that is not part of
 an Auto Scaling group. This change may result in CloudFormation drift for any stacks
 that have this resource.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ReplaceInstanceProfileV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ReplaceInstanceProfileV2"
      ],
      "default": "AWSManagedServices-ReplaceInstanceProfileV2"
    },
    "Region": {
```

```
      "description": "The AWS Region where the EC2 instance is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the EC2 instance, in the form
 i-1234567890abcdef0.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "InstanceProfile": {
          "description": "An IAM instance profile name defined in your account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[\\w+=,.@-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "InstanceProfile"
        ]
      },
      "required": [
        "InstanceId",
        "InstanceProfile"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-37qquo9wbpa8x

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Delete or deactivate access key

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete or Deactivate Access Key",
  "description": "Delete or deactivate the specified AWS IAM access key ID for the
 specified user.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeactivateIAMAccessKey.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeactivateIAMAccessKey"
      ],
      "default": "AWSManagedServices-DeactivateIAMAccessKey"
    },
    "Region": {
      "description": "The AWS Region of the account.",
      "type": "string",
      "enum": [
        "us-east-1",
```

```
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "UserName": {
          "description": "The name of the IAM user that the key belongs to.",
          "type": "string",
          "pattern": "^[\\w+=,.@-]+",
          "minLength": 1,
          "maxLength": 128
        },
        "AccessKeyId": {
          "description": "The ID of the access key to delete or deactivate.",
          "type": "string",
          "pattern": "^AKIA\\w+$",
          "minLength": 16,
          "maxLength": 128
        },
```

```
        "Delete": {
          "description": "True to delete the access key for the specified user, False
 to deactivate it without deleting.",
          "type": "boolean",
          "default": false
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "UserName",
          "AccessKeyId",
          "Delete"
        ]
      },
      "required": [
        "UserName",
        "AccessKeyId",
        "Delete"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-37vqa0oggka3q

Classifications:

- Management | Advanced stack components | RDS database stack | Stop Aurora cluster

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop Aurora DB Cluster",
  "description": "Stop an Aurora DB cluster, which is a provisioned capacity type
 and does not have cross-region read replicas. The cluster must be in the 'available'
 state.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StopDBCluster.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StopDBCluster"
      ],
      "default": "AWSManagedServices-StopDBCluster"
    },
    "Region": {
      "description": "The AWS Region where the cluster is.",
      "type": "string",
      "enum": [
        "us-east-1",
        "us-east-2",
        "us-west-1",
        "us-west-2",
        "eu-west-1",
        "eu-west-2",
        "eu-west-3",
        "eu-south-1",
        "eu-north-1",
        "eu-central-1",
        "ca-central-1",
        "ap-southeast-1",
        "ap-southeast-2",
        "ap-southeast-3",
        "ap-south-1",
        "ap-northeast-1",
        "ap-northeast-2",
        "ap-northeast-3",
        "ap-east-1",
        "sa-east-1",
        "me-south-1",
        "af-south-1",
        "us-gov-west-1",
```

```
        "us-gov-east-1",
        "cn-northwest-1",
        "cn-north-1"
      ]
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBClusterIdentifier": {
          "description": "The unique RDS DB cluster identifier.",
          "type": "string",
          "pattern": "^[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$|^$"
        }
      },
      "metadata": {
        "ui:order": [
          "DBClusterIdentifier"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBClusterIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-38s4s4tm4ic4u

Classifications:

- Management | Advanced stack components | EC2 instance stack | Update


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update EC2 stack",
  "description": "Use to modify the properties of an EC2 instance created using CT id
 ct-14027q0sjyt1h, version 3.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC that contains the EC2 Instance, in the form
 vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the EC2 instance that you are updating, in the
 form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the EC2 instance.",
      "type": "object",
      "properties": {
        "InstanceDetailedMonitoring": {
          "description": "True to enable detailed monitoring on the instance, false to
 use only basic monitoring.",
          "type": "boolean"
        },
        "InstanceEBSOptimized": {
          "description": "True for the instance to be optimized for Amazon Elastic
 Block Store I/O, false for it to not be. If you set this to true, choose an
 InstanceType that supports EBS optimization. Updates will stop and start Amazon EBS-
backed instances.",
          "type": "boolean"
        },
        "InstanceProfile": {
```

```
          "description": "An IAM instance profile name defined in your account for the
EC2 instance.",
          "type": "string",
          "minLength": 1,
          "maxLength": 128,
          "pattern": "^customer[\\w-]{1,120}$"
        },
        "InstanceType": {
          "description": "The type of EC2 instance to deploy. If InstanceEBSOptimized
= true, specify an InstanceType that supports EBS optimization. Changing the instance
type will result in instance stop and start.",
          "type": "string"
        },
        "InstanceUserData": {
          "description": "A newline-delimited string where each line is part of the
script to be run on boot. Changing the UserData will result in instance stop and
start. Note: Existing instances do not pick up changes in UserData automatically,
in order for the instance to execute modified UserData you must perform additional
changes by logging in to the instance.",
          "type": "string",
          "maxLength": 4096
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "InstanceDetailedMonitoring",
          "InstanceEBSOptimized",
          "InstanceProfile",
          "InstanceType",
          "InstanceUserData"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
```

```
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-38xcr0q86k9lh

Classifications:

- Deployment | Managed landing zone | Management account | Create developer mode account (with VPC)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Developer Mode Account With VPC",
  "description": "Create a managed AWS landing zone developer mode account and a VPC
 with up to 10 private subnets and up to 5 optional public subnets per availability
 zone (AZ) for two or three AZ's. Optionally, also create an AWS Backup plan with up to
 four different rules. Managed AWS landing zone core accounts must already be onboarded
 to AWS Managed Services (AMS).",
  "type": "object",
  "properties": {
    "AccountName": {
      "description": "A name for the new developer mode account. Max length 50
 characters. The underscore (_) is not allowed.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9]{1}[a-zA-Z0-9.-]{0,49}$"
    },
    "AccountEmail": {
      "description": "The email address for the new developer mode account. The email
 must be unique per developer mode account.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
    },
    "DeveloperModeOUName": {
      "description": "The name of an existing organizational unit (OU) for this
 developer mode account, in the form of <developer mode ou name>:<child ou name>. The
 default value is applications:development.",
      "type": "string",
      "default": "applications:development"
```

```
    },
    "SupportLevel": {
      "description": "The account's AMS support level, Premium or Plus.",
      "type": "string",
      "enum": [
        "plus",
        "premium"
      ]
    },
    "VpcName": {
      "description": "A meaningful name for the developer mode account VPC. Must be
 unique within this developer mode account.",
      "type": "string"
    },
    "NumberOfAZs": {
      "description": "The number of availability zones (AZs) that the VPC supports.
 Options are 2 or 3.",
      "type": "number",
      "minimum": 2,
      "maximum": 3
    },
    "VpcCIDR": {
      "description": "The Classless Inter-Domain Routing (CIDR) for the VPC.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "RouteType": {
      "description": "The AWS Transit Gateway application route table connection type.
 For this VPC to accept connections from other VPCs, use routable. For it to not accept
 those connections, use isolated. The default is routable.",
      "type": "string",
      "enum": [
        "isolated",
        "routable"
      ],
      "default": "routable"
    },
    "TransitGatewayApplicationRouteTableName": {
      "description": "The existing AWS Transit Gateway route table for this developer
 mode account VPC. The default is defaultAppRouteDomain. To create a new application
 route table, use the Create Application Route Table change type.",
      "type": "string",
      "default": "defaultAppRouteDomain"
```

```
    },
    "PublicSubnetAZ1CIDR": {
      "description": "The CIDR for the optional first public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnetAZ2CIDR": {
      "description": "The CIDR for the optional first public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnetAZ3CIDR": {
      "description": "The CIDR for the optional first public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ1CIDR": {
      "description": "The CIDR for the optional second public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ2CIDR": {
      "description": "The CIDR for the optional second public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet2AZ3CIDR": {
      "description": "The CIDR for the optional second public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ1CIDR": {
```

```
      "description": "The CIDR for the optional third public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ2CIDR": {
      "description": "The CIDR for the optional third public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet3AZ3CIDR": {
      "description": "The CIDR for the optional third public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ1CIDR": {
      "description": "The CIDR for the optional fourth public subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ2CIDR": {
      "description": "The CIDR for the optional fourth public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet4AZ3CIDR": {
      "description": "The CIDR for the optional fourth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ1CIDR": {
      "description": "The CIDR for the optional fifth public subnet in availability
 zone 1.",
```

```
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ2CIDR": {
      "description": "The CIDR for the optional fifth public subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PublicSubnet5AZ3CIDR": {
      "description": "The CIDR for the optional fifth public subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ1CIDR": {
      "description": "The CIDR for the first private subnet in availability zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ2CIDR": {
      "description": "The CIDR for the first private subnet in availability zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet1AZ3CIDR": {
      "description": "The CIDR for the first private subnet in optional availability
 zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet2AZ1CIDR": {
      "description": "The CIDR for the optional second private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
```

```
    "PrivateSubnet2AZ2CIDR": {
      "description": "The CIDR for the optional second private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet2AZ3CIDR": {
      "description": "The CIDR for the optional second private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ1CIDR": {
      "description": "The CIDR for the optional third private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ2CIDR": {
      "description": "The CIDR for the optional third private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet3AZ3CIDR": {
      "description": "The CIDR for the optional third private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ1CIDR": {
      "description": "The CIDR for the optional fourth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ2CIDR": {
```

```
      "description": "The CIDR for the optional fourth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet4AZ3CIDR": {
      "description": "The CIDR for the optional fourth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ1CIDR": {
      "description": "The CIDR for the optional fifth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ2CIDR": {
      "description": "The CIDR for the optional fifth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet5AZ3CIDR": {
      "description": "The CIDR for the optional fifth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ1CIDR": {
      "description": "The CIDR for the optional sixth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ2CIDR": {
      "description": "The CIDR for the optional sixth private subnet in availability
 zone 2.",
```

```
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet6AZ3CIDR": {
      "description": "The CIDR for the optional sixth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ1CIDR": {
      "description": "The CIDR for the optional seventh private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ2CIDR": {
      "description": "The CIDR for the optional seventh private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet7AZ3CIDR": {
      "description": "The CIDR for the optional seventh private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ1CIDR": {
      "description": "The CIDR for the optional eighth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ2CIDR": {
      "description": "The CIDR for the optional eighth private subnet in availability
 zone 2.",
      "type": "string",
```

```
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet8AZ3CIDR": {
      "description": "The CIDR for the optional eighth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ1CIDR": {
      "description": "The CIDR for the optional ninth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ2CIDR": {
      "description": "The CIDR for the optional ninth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet9AZ3CIDR": {
      "description": "The CIDR for the optional ninth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet10AZ1CIDR": {
      "description": "The CIDR for the optional tenth private subnet in availability
 zone 1.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "PrivateSubnet10AZ2CIDR": {
      "description": "The CIDR for the optional tenth private subnet in availability
 zone 2.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
```

```
    },
    "PrivateSubnet10AZ3CIDR": {
      "description": "The CIDR for the optional tenth private subnet in optional
 availability zone 3. Only required if three availability zones are chosen.",
      "type": "string",
      "pattern": "^(([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]|[1-9]
[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2]))$"
    },
    "DirectAlertsEmail": {
      "description": "Email address to receive specifically tagged resource-based
 alerts, and the onboarding process will create your SNS subscription. If not
 specified, then you can subscribe later using the DirectCustomerAlerts change type
 (ct-t-3rcl9u1k017wu).",
      "type": "string",
      "pattern": "^[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-
zA-Z0-9])?(?:\\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)*$"
    },
    "SamlMetadataDocumentURL": {
      "description": "The URL that points to the Security Assertion Markup
 Language(SAML) metadata document that is used to enable federated access to the
 developer mode account. Typically, a pre-signed URL for an Amazon S3 object.",
      "type": "string",
      "pattern": "^https://.+$|^$|s3://.+$"
    },
    "BackupPlanName": {
      "type": "string",
      "description": "A meaningful name for the AWS Backup plan, which is a policy
 expression that defines when and how you want to back up your AWS resources.",
      "default": "default-backup-plan"
    },
    "ResourceTagKey": {
      "type": "string",
      "description": "The tag key (case sensitive) of the resources to be backed up.
 For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
      "default": "Backup"
    },
    "ResourceTagValue": {
      "type": "string",
      "description": "The tag value (case sensitive) of the resources to be backed up.
 For example, if you want to use a tag key:value pair like 'Department:accounting',
 you need to provide 'Department' as the ResourceTagKey and 'accounting' as the
 ResourceTagValue.",
```

```
      "default": "True"
    },
    "BackupRule1ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$",
      "default": "cron(0 2 ? * * )"
    },
    "BackupRule1DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the daily backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 7
    },
    "BackupRule1MoveToColdStorageAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that the daily backup is moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
    "BackupRule2ScheduleExpression": {
      "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
      "type": "string",
      "pattern": "^(cron|rate)\\(.*\\)$"
    },
    "BackupRule2DeleteAfterDays": {
      "type": "integer",
      "description": "The number of days after creation that weekly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
      "minimum": 0,
      "maximum": 35600,
      "default": 0
    },
```

```
   "BackupRule2MoveToColdStorageAfterDays": {
     "type": "integer",
     "description": "The number of days after creation that weekly backups are moved
to cold storage. Valid values are between 1 and 35600. If the value is set to 0, the
backup never moves to cold storage.",
     "minimum": 0,
     "maximum": 35600,
     "default": 0
   },
   "BackupRule3ScheduleExpression": {
     "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
     "type": "string",
     "pattern": "^(cron|rate)\\(.*\\)$"
   },
   "BackupRule3DeleteAfterDays": {
     "type": "integer",
     "description": "The number of days after creation that monthly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
     "minimum": 0,
     "maximum": 35600,
     "default": 0
   },
   "BackupRule3MoveToColdStorageAfterDays": {
     "type": "integer",
     "description": "The number of days after creation that the monthly backups are
moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
the backup never moves to cold storage.",
     "minimum": 0,
     "maximum": 35600,
     "default": 0
   },
   "BackupRule4ScheduleExpression": {
     "description": "A cron expression that specifies when the AWS Backup service
initiates a backup job. For example, cron(0 2 ? * * *) will set a daily backup for 2am
UTC time.",
     "type": "string",
     "pattern": "^(cron|rate)\\(.*\\)$"
   },
   "BackupRule4DeleteAfterDays": {
     "type": "integer",
```

```
     "description": "The number of days after creation that the yearly backups are
deleted. Valid values are between 1 and 35600. If a value is set to 0, the backup
never expires.",
     "minimum": 0,
     "maximum": 35600,
     "default": 0
   },
   "BackupRule4MoveToColdStorageAfterDays": {
     "type": "integer",
     "description": "The number of days after creation that the yearly backups are
moved to cold storage. Valid values are between 1 and 35600. If the value is set to 0,
the backup never moves to cold storage.",
     "minimum": 0,
     "maximum": 35600,
     "default": 0
   },
   "PatchOrchestratorFirstTagKey": {
     "description": "The first tag-key to use for creating your \"Patch Group\" tag
values. For example, AppId. Specify null if you already have defined your own patch
groups with a \"Patch Group\" tag.",
     "type": "string",
     "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$"
   },
   "PatchOrchestratorSecondTagKey": {
     "description": "The second tag-key to use for creating your \"Patch Group\" tag
values. For example, Environment. Specify null if you already have defined your own
patch groups with a \"Patch Group\" tag.",
     "type": "string",
     "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$"
   },
   "PatchOrchestratorThirdTagKey": {
     "description": "The third tag-key to use for creating your \"Patch Group\" tag
values. For example, Group. Specify null if you already have defined your own patch
groups with a \"Patch Group\" tag.",
     "type": "string",
     "pattern": "^[a-zA-Z0-9+\\-=._:/@ ]{1,128}$",
     "default": "null"
   },
   "PatchOrchestratorDefaultMaintenanceWindowCutoff": {
     "description": "The number of hours before the end of the Default Maintenance
Window in which no new patching commands are started. This interval exists to allow
enough time for patching to complete before the window ends.",
     "default": 0,
     "minimum": 0,
```

```
      "maximum": 23,
      "type": "integer"
    },
    "PatchOrchestratorDefaultMaintenanceWindowDuration": {
      "description": "The duration of the maintenance window in hours.",
      "default": 4,
      "minimum": 1,
      "maximum": 24,
      "type": "integer"
    },
    "PatchOrchestratorDefaultMaintenanceWindowSchedule": {
      "description": "The schedule of the maintenance window in the form of a cron or
rate expression. For example cron(0 18 * * ? *) would create a window at 18:00 every
day, and rate(7 days) would create a window every seven days.",
      "default": "cron(0 18 * * ? *)",
      "minLength": 1,
      "maxLength": 256,
      "pattern": "^cron\\([0-9a-zA-Z\\ ?*#-,\\/]+\\)$|^rate\\([0-9a-zA-Z\\ ]+\\)$",
      "type": "string"
    },
    "PatchOrchestratorDefaultMaintenanceWindowTimeZone": {
      "description": "The time zone that the scheduled maintenance window executions
are based on, in Internet Assigned Numbers Authority (IANA) format.",
      "default": "UTC",
      "pattern": "^[a-zA-Z_]+(\\+|/)?[a-zA-Z0-9_-]*(\\+|/)?[a-zA-Z0-9_-]+$",
      "type": "string"
    },
    "PatchOrchestratorDefaultPatchBackupRetentionInDays": {
      "description": "The number of days the backup taken before patching will remain
available.",
      "default": 60,
      "minimum": 1,
      "maximum": 90,
      "type": "integer"
    },
    "PatchOrchestratorNotificationEmails": {
      "description": "One or more email addresses to receive notifications about
default patching status. Use group distribution lists instead of individual emails.",
      "items": {
        "type": "string",
        "pattern": "^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$"
      },
      "minItems": 1,
      "maxItems": 5,
```

```
      "type": "array",
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "AccountName",
      "AccountEmail",
      "DeveloperModeOUName",
      "SupportLevel",
      "DirectAlertsEmail",
      "SamlMetadataDocumentURL",
      "VpcName",
      "VpcCIDR",
      "NumberOfAZs",
      "RouteType",
      "TransitGatewayApplicationRouteTableName",
      "PublicSubnetAZ1CIDR",
      "PublicSubnetAZ2CIDR",
      "PublicSubnetAZ3CIDR",
      "PublicSubnet2AZ1CIDR",
      "PublicSubnet2AZ2CIDR",
      "PublicSubnet2AZ3CIDR",
      "PublicSubnet3AZ1CIDR",
      "PublicSubnet3AZ2CIDR",
      "PublicSubnet3AZ3CIDR",
      "PublicSubnet4AZ1CIDR",
      "PublicSubnet4AZ2CIDR",
      "PublicSubnet4AZ3CIDR",
      "PublicSubnet5AZ1CIDR",
      "PublicSubnet5AZ2CIDR",
      "PublicSubnet5AZ3CIDR",
      "PrivateSubnet1AZ1CIDR",
      "PrivateSubnet1AZ2CIDR",
      "PrivateSubnet1AZ3CIDR",
      "PrivateSubnet2AZ1CIDR",
      "PrivateSubnet2AZ2CIDR",
      "PrivateSubnet2AZ3CIDR",
      "PrivateSubnet3AZ1CIDR",
      "PrivateSubnet3AZ2CIDR",
      "PrivateSubnet3AZ3CIDR",
      "PrivateSubnet4AZ1CIDR",
      "PrivateSubnet4AZ2CIDR",
      "PrivateSubnet4AZ3CIDR",
```

```
      "PrivateSubnet5AZ1CIDR",
      "PrivateSubnet5AZ2CIDR",
      "PrivateSubnet5AZ3CIDR",
      "PrivateSubnet6AZ1CIDR",
      "PrivateSubnet6AZ2CIDR",
      "PrivateSubnet6AZ3CIDR",
      "PrivateSubnet7AZ1CIDR",
      "PrivateSubnet7AZ2CIDR",
      "PrivateSubnet7AZ3CIDR",
      "PrivateSubnet8AZ1CIDR",
      "PrivateSubnet8AZ2CIDR",
      "PrivateSubnet8AZ3CIDR",
      "PrivateSubnet9AZ1CIDR",
      "PrivateSubnet9AZ2CIDR",
      "PrivateSubnet9AZ3CIDR",
      "PrivateSubnet10AZ1CIDR",
      "PrivateSubnet10AZ2CIDR",
      "PrivateSubnet10AZ3CIDR",
      "BackupPlanName",
      "ResourceTagKey",
      "ResourceTagValue",
      "BackupRule1ScheduleExpression",
      "BackupRule1DeleteAfterDays",
      "BackupRule1MoveToColdStorageAfterDays",
      "BackupRule2ScheduleExpression",
      "BackupRule2DeleteAfterDays",
      "BackupRule2MoveToColdStorageAfterDays",
      "BackupRule3ScheduleExpression",
      "BackupRule3DeleteAfterDays",
      "BackupRule3MoveToColdStorageAfterDays",
      "BackupRule4ScheduleExpression",
      "BackupRule4DeleteAfterDays",
      "BackupRule4MoveToColdStorageAfterDays",
      "PatchOrchestratorFirstTagKey",
      "PatchOrchestratorSecondTagKey",
      "PatchOrchestratorThirdTagKey",
      "PatchOrchestratorDefaultMaintenanceWindowCutoff",
      "PatchOrchestratorDefaultMaintenanceWindowDuration",
      "PatchOrchestratorDefaultMaintenanceWindowSchedule",
      "PatchOrchestratorDefaultMaintenanceWindowTimeZone",
      "PatchOrchestratorDefaultPatchBackupRetentionInDays",
      "PatchOrchestratorNotificationEmails"
    ]
  },
```

```
  "additionalProperties": false,
  "required": [
    "AccountName",
    "AccountEmail",
    "SupportLevel",
    "VpcName",
    "VpcCIDR",
    "NumberOfAZs",
    "PrivateSubnet1AZ1CIDR",
    "PrivateSubnet1AZ2CIDR",
    "BackupPlanName",
    "ResourceTagKey",
    "ResourceTagValue",
    "BackupRule1ScheduleExpression"
  ]
}
```

Schema for Change Type ct-3929xwf222jri

Classifications:

- Management | Advanced stack components | Network Load Balancer | Remove listener certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove NLB Listener Certificate",
  "description": "Remove a certificate from the specified Network Load Balancer (NLB)
 listener. Use the RemediateStackDrift parameter for the automation to try to remediate
 drift, if it is introduced.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RemoveCertificateFromElbv2Listener.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveCertificateFromElbv2Listener"
      ],
      "default": "AWSManagedServices-RemoveCertificateFromElbv2Listener"
    },
    "Region": {
      "description": "The AWS Region where the network load balancer listener is
 located, in the form us-east-1.",
```

```
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ListenerArn": {
          "description": "The Amazon Resource Name (ARN) of the listener in
 the form arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/net/
sample/1234567890abcdfe/1234567890abcdfe.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):elasticloadbalancing:[a-z]{2}-[a-
z]+-[0-9]{1}:[0-9]{12}:listener/net/[A-Za-z0-9-]+/[a-z0-9-]+/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CertificateArn": {
          "description": "The Amazon Resource Name (ARN) of the certificate in the form
 arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused by
 removing the certificate from the Loadbalancer Listener. False to not attempt drift
 remediation. Drift remediation can be performed only on CloudFormation stacks that
 were created using a CT other than the Ingestion CT ct-36cn2avfrrj9v and that are
 in sync with the definitions in the stack template prior to removing the certificate
 from the Loadbalancer Listener. Set to False to remove the certificate from the
 Loadbalancer Listener in an ingested stack if any drift introduced by the change is
 acceptable.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ListenerArn",
          "CertificateArn",
          "RemediateStackDrift"
        ]
      },
      "additionalProperties": false,
      "required": [
        "CertificateArn",
        "ListenerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-393q3yaq9ewlm

Classifications:

- Deployment | Advanced stack components | RDS snapshot | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create RDS DB Snapshot",
  "description": "Create a snapshot of an Amazon Relational Database Service (RDS)
 database (DB) instance. The snapshot will be encrypted with the same KMS key as the DB
 instance, or unencrypted if DB instance is unencrypted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateDBSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateDBSnapshot"
      ],
      "default": "AWSManagedServices-CreateDBSnapshot"
    },
    "Region": {
      "description": "The AWS Region in which the RDS DB is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DBInstanceIdentifier": {
          "description": "The identifier for the RDS DB that you are creating a
 snapshot of.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,62}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
```

```
        "DBSnapshotName": {
          "description": "A name for the DB snapshot.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z][a-zA-Z0-9-]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "DBInstanceIdentifier",
          "DBSnapshotName"
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBInstanceIdentifier",
        "DBSnapshotName"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-39c5qiasbe4he

Classifications:

- Management | Advanced stack components | Redshift | Resume cluster

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Resume Redshift Cluster",
  "description": "Resume a paused Amazon Redshift cluster.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ResumeRedshiftCluster.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ResumeRedshiftCluster"
      ],
      "default": "AWSManagedServices-ResumeRedshiftCluster"
    },
    "Region": {
      "description": "The AWS Region in which the Amazon Redshift cluster is located,
 in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterIdentifier": {
          "description": "The Amazon Redshift cluster identifier. For example, myred-
cluster-1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!(ams-|mc-))[a-z]+(-?[a-z0-9]+)+$",
            "minLength": 1,
            "maxLength": 63
          },
          "minItems": 1,
          "maxItems": 1
        }
```

```
      },
      "metadata": {
        "ui:order": [
          "ClusterIdentifier"
        ]
      },
      "additionalProperties": false,
      "required": [
        "ClusterIdentifier"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3cp96z7r065e4

Classifications:

- Management | Advanced stack components | Security group | Delete (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete or disassociate a security group",
  "description": "Disassociate a security group from the specified AWS resources and
 optionally delete the security group.",
  "type": "object",
  "properties": {
    "SecurityGroupId": {
```

```
      "description": "ID of the security group to be deleted or disassociated from AWS
resources. You cannot delete a security group that is still associated with any AWS
resources.",
      "type": "string",
      "pattern": "^sg-[0-9a-zA-Z]{8}$|^sg-[0-9a-zA-Z]{17}$"
    },
    "DisassociatedResources": {
      "description": "AWS resources to disassociate the security group from. For
example, EC2 instance IDs, RDS DB instance IDs, Load Balancer names, DSM replication
instance names, EFS mount target IDs, ElastiCache cluster IDs.",
      "type": "array",
      "items": {
        "type": "string",
        "minLength": 1,
        "maxLength": 64
      },
      "minItems": 0,
      "maxItems": 10,
      "uniqueItems": true
    },
    "DeleteSecurityGroup": {
      "description": "True if the security should be deleted in addition to
disassociating it from the AWS resources, or false if not. Default is false.",
      "type": "boolean",
      "default": false
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "SecurityGroupId",
      "DisassociatedResources",
      "DeleteSecurityGroup",
      "Priority"
```

```
    ]
  },
  "required": [
    "SecurityGroupId",
    "DisassociatedResources",
    "DeleteSecurityGroup"
  ]
}
```

Schema for Change Type ct-3cx7we852p3af

Classifications:

- Deployment | Advanced stack components | Tag | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Resource Tags",
  "description": "Add tags to existing, supported resources: Autoscaling, EC2,
 Elastic Load Balancing, RDS, S3 buckets and Redshift clusters. Additionally,
 CloudWatch LogGroups that do not belong to a CloudFormation stack are supported. AMS
 infrastructure stacks (stacks named mc-*) cannot have tags added with this change
 type.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateTags.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateTags"
      ],
      "default": "AWSManagedServices-UpdateTags"
    },
    "Region": {
      "description": "The AWS Region where the resources to be tagged are, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
        "ResourceArns": {
          "description": "A list of up to 50 Amazon resource names (ARNs), or the
 resource IDs, of the resources to be tagged. Use resource ID only for these resource
 types: EC2 instance, EBS volume, EBS snapshot, AMI, and security group. Use the full
 ARN for all other supported resource types.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:(autoscaling|ec2|elasticloadbalancing|logs|rds|s3|
redshift):(|[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}):(|[0-9]{12}):.*)$|^(ami|i|vol|
sg|snap)-([a-f0-9]{8}|[a-f0-9]{17})$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "AddOrUpdateTags": {
          "description": "Up to fifty tags (key/value pairs) to categorize the
 resource, in the form {\"Key\":\"TagKey1\",\"Value\":\"TagValue1\"}. If the tag
 exists, the value for it is overwritten. If the tag does not exist, it is added to the
 resource. Characters allowed in tags can vary by AWS service. For information about
 what characters can be used to tag resources in a particular AWS service, please refer
 to its documentation. In general, allowed characters in tags are letters, numbers,
 spaces and the following characters: _ . : / = + - @.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^\\{\\}$|^\\{\"Key\":\"((aws-migration-project-id)|(?![aA]
[mMwW][sS])[\\x00-\\x7F+]{1,128})\",\"Value\":\"[\\x00-\\x7F+]{0,255}\"\\}"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "ResourceArns",
          "AddOrUpdateTags"
        ]
      },
      "required": [
        "ResourceArns",
        "AddOrUpdateTags"
```

```
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "DocumentName",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3d0lrfb8eckuu

Classifications:

- Management | Directory Service | Computer object | Remove

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remove Computer Object",
  "description": "Remove a stale computer object from Microsoft Active Directory
 (AD) and the corresponding DNS A and PTR records from DNS. Removing the computer
 object will prevent anyone from raising access against this host using the AMS access
 control. For multi-account landing zone (MALZ), use this change type in the shared
 services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-RemoveADComputerObject-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-RemoveADComputerObject-Admin"
      ],
      "default": "AWSManagedServices-RemoveADComputerObject-Admin"
```

```
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Hostname": {
          "description": "The hostname of the computer object in Active Directory.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\-]{1,15}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Hostname"
        ]
      },
      "required": [
        "Hostname"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
```

```
  "additionalProperties": false
}
```

Schema for Change Type ct-3dfnglm4ombbs

Classifications:

- Deployment | Monitoring and notification | SNS | Create (topic and subscription)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SNS topic",
  "description": "Create an SNS topic and up to five subscriptions.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
```

```
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-eakrsalqo9m62tpun",
      "type": "string",
      "enum": [
        "stm-eakrsalqo9m62tpun"
      ],
      "default": "stm-eakrsalqo9m62tpun"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
```

```
      "properties": {
        "TopicName": {
          "type": "string",
          "description": "A name for the SNS topic. If not specified, a unique topic
 name is generated. Name can contain up to 256 alphanumeric, - and _ characters.",
          "pattern": "^[a-zA-Z0-9-_]{0,256}|^$",
          "default": ""
        },
        "DisplayName": {
          "type": "string",
          "description": "A display name for the SNS topic for use with short message
 service (SMS) messages. Must contain up to 10 alphanumeric, - and _ characters.",
          "pattern": "^[a-zA-Z0-9-_]{0,10}|^$",
          "default": ""
        },
        "Subscription1Protocol": {
          "type": "string",
          "description": "The Endpoint Protocol for the Subscription1Endpoint
 parameter.",
          "enum": [
            "http",
            "https",
            "email",
            "sqs",
            "sms",
            "lambda"
          ]
        },
        "Subscription1Endpoint": {
          "type": "string",
          "description": "One of the AMS supported valid endpoints: SQS, SMS,
 Email, HTTP, HTTPS and Lambda to subscribe to this topic. For details, refer to AWS
 documentation for valid SNS topic subscription endpoints.",
          "pattern": "^http://.*$|^https://.*$|^(arn:(aws|aws-us-gov):(sqs|lambda):[a-
z0-9-]+:[0-9]{12}:.+)$|^\\+[0-9]*$|^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$|^$",
          "default": ""
        },
        "Subscription1RawMessageDelivery": {
          "type": "string",
          "description": "True to enable raw message delivery (messages are not encoded
 in JSON that provides metadata), false to not. Use only for SQS, HTTP or HTTPS
 endpoint.",
          "enum": [
            "true",
```

```
            "false"
          ],
          "default": "false"
        },
        "Subscription2Protocol": {
          "type": "string",
          "description": "The Endpoint Protocol for the Subscription2Endpoint
 parameter.",
          "enum": [
            "http",
            "https",
            "email",
            "sqs",
            "sms",
            "lambda"
          ]
        },
        "Subscription2Endpoint": {
          "type": "string",
          "description": "One of the AMS supported valid endpoints: SQS, SMS,
 Email, HTTP, HTTPS and Lambda to subscribe to this topic. For details, refer to AWS
 documentation for valid SNS topic subscription endpoints.",
          "pattern": "^http://.*$|^https://.*$|^(arn:(aws|aws-us-gov):(sqs|lambda):[a-
z0-9-]+:[0-9]{12}:.+)$|^\\+[0-9]*$|^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$|^$",
          "default": ""
        },
        "Subscription2RawMessageDelivery": {
          "type": "string",
          "description": "True to enable raw message delivery (messages are not encoded
 in JSON that provides metadata), false to not. Use only for SQS, HTTP or HTTPS
 endpoint.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "Subscription3Protocol": {
          "type": "string",
          "description": "The Endpoint Protocol for the Subscription3Endpoint
 parameter.",
          "enum": [
            "http",
            "https",
```

```
            "email",
            "sqs",
            "sms",
            "lambda"
          ]
        },
        "Subscription3Endpoint": {
          "type": "string",
          "description": "One of the AMS supported valid endpoints: SQS, SMS,
 Email, HTTP, HTTPS and Lambda to subscribe to this topic. For details, refer to AWS
 documentation for valid SNS topic subscription endpoints.",
          "pattern": "^http://.*$|^https://.*$|^(arn:(aws|aws-us-gov):(sqs|lambda):[a-
z0-9-]+:[0-9]{12}:.+)$|^\\+[0-9]*$|^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$|^$",
          "default": ""
        },
        "Subscription3RawMessageDelivery": {
          "type": "string",
          "description": "True to enable raw message delivery (messages are not encoded
 in JSON that provides metadata), false to not. Use only for SQS, HTTP or HTTPS
 endpoint.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "Subscription4Protocol": {
          "type": "string",
          "description": "The Endpoint Protocol for the Subscription4Endpoint
 parameter.",
          "enum": [
            "http",
            "https",
            "email",
            "sqs",
            "sms",
            "lambda"
          ]
        },
        "Subscription4Endpoint": {
          "type": "string",
          "description": "One of the AMS supported valid endpoints: SQS, SMS,
 Email, HTTP, HTTPS and Lambda to subscribe to this topic. For details, refer to AWS
 documentation for valid SNS topic subscription endpoints.",
```

```
          "pattern": "^http://.*$|^https://.*$|^(arn:(aws|aws-us-gov):(sqs|lambda):[a-
z0-9-]+:[0-9]{12}:.+)$|^\\+[0-9]*$|^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$|^$",
          "default": ""
        },
        "Subscription4RawMessageDelivery": {
          "type": "string",
          "description": "True to enable raw message delivery (messages are not encoded
 in JSON that provides metadata), false to not. Use only for SQS, HTTP or HTTPS
 endpoint.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "Subscription5Protocol": {
          "type": "string",
          "description": "The Endpoint Protocol for the Subscription5Endpoint
 parameter.",
          "enum": [
            "http",
            "https",
            "email",
            "sqs",
            "sms",
            "lambda"
          ]
        },
        "Subscription5Endpoint": {
          "type": "string",
          "description": "One of the AMS supported valid endpoints: SQS, SMS,
 Email, HTTP, HTTPS and Lambda to subscribe to this topic. For details, refer to AWS
 documentation for valid SNS topic subscription endpoints.",
          "pattern": "^http://.*$|^https://.*$|^(arn:(aws|aws-us-gov):(sqs|lambda):[a-
z0-9-]+:[0-9]{12}:.+)$|^\\+[0-9]*$|^[a-zA-Z0-9-_+.]+@[a-zA-Z0-9-_+.]+$|^$",
          "default": ""
        },
        "Subscription5RawMessageDelivery": {
          "type": "string",
          "description": "True to enable raw message delivery (messages are not encoded
 in JSON that provides metadata), false to not. Use only for SQS, HTTP or HTTPS
 endpoint.",
          "enum": [
            "true",
```

```
            "false"
          ],
          "default": "false"
        },
        "KmsMasterKeyId": {
          "type": "string",
          "description": "A valid AWS KMS key ARN to enable server-side
 encryption at rest, in the form of 'arn:aws:kms:ap-southeast-2:123456789023:key/
bb43bd18-3a75-482e-822d-d0d3a5544dc8'",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "TopicName",
          "DisplayName",
          "KmsMasterKeyId",
          "Subscription1Protocol",
          "Subscription1Endpoint",
          "Subscription1RawMessageDelivery",
          "Subscription2Protocol",
          "Subscription2Endpoint",
          "Subscription2RawMessageDelivery",
          "Subscription3Protocol",
          "Subscription3Endpoint",
          "Subscription3RawMessageDelivery",
          "Subscription4Protocol",
          "Subscription4Endpoint",
          "Subscription4RawMessageDelivery",
          "Subscription5Protocol",
          "Subscription5Endpoint",
          "Subscription5RawMessageDelivery"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
```

```
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Name",
    "Description",
    "VpcId",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3dfubbpesm2v9

Classifications:

- Management | Standalone resources | EC2 instance | Terminate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Terminate EC2 Instances",
  "description": "Terminate up to fifty EC2 instances. The automation checks that
 none of the instances are part of an Auto Scaling group and none have termination
 protection enabled. Instances meeting either of those criteria are not terminated.
 Standalone resources for testing purposes are created by AMS upon your request, they
 are not part of a stack and can't be deleted with ct-0q0bic0ywqk6c.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-TerminateStandaloneInstances.",
      "type": "string",
      "enum": [
        "AWSManagedServices-TerminateStandaloneInstances"
      ],
      "default": "AWSManagedServices-TerminateStandaloneInstances"
    },
    "Region": {
      "description": "The AWS Region where the instances are located, in the form us-
east-1.",
```

```
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Confirmation": {
      "description": "Explicitly confirm the termination of the specified EC2 instances
with 'terminate instances', note that the RFC is not created if this parameter
is null. Additionally, note that Amazon EBS volumes with DeleteOnTermination=true
are automatically deleted when the instance terminates; for the root volume of an
instance, DeleteOnTermination=true by default.",
      "type": "string",
      "pattern": "^terminate instances$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceIds": {
          "description": "A list of up to fifty EC2 instance IDs, in the form
i-1234567890abcdef0 or i-a123456b.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Confirmation",
```

```
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Confirmation",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3dgbnh6gpst4d

Classifications:

- Management | Standard stacks | Stack | Stop

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop stack",
  "description": "Use to stop all running EC2 instances in the specified stack.",
  "additionalProperties": false,
  "type": "object",
  "properties": {
    "StackId": {
      "pattern": "^stack-[a-z0-9]{17}$",
      "description": "ID of the stack to stop, in the form stack-a1b2c3d4e5f67890e. All
 running EC2 instances in the stack will be stopped.",
      "type": "string"
    }
  },
  "required": [
    "StackId"
  ]
}
```

Schema for Change Type ct-3dpd8mdd9jn1r

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create entity or policy (review required)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create IAM Resource",
  "description": "Create Identity and Access Management (IAM) user, role, or policy.",
  "type": "object",
  "properties": {
    "UseCase": {
      "description": "Provide a detailed use case for the IAM user, role, or policy.
 Note that IAM users are recommended when long-term credentials are required, otherwise
 IAM roles are recommended.",
      "type": "string",
      "minLength": 1,
      "maxLength": 1000
    },
    "IAM User": {
      "description": "Create IAM User.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "UserName": {
            "description": "A name for the IAM user. The name can be up to 64
 characters in length, and is limited to use characters a-z, A-Z, 0-9, and _+=,.@-.",
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$",
            "minLength": 1,
            "maxLength": 64
          },
          "AccessType": {
            "description": "How the user will access AWS.",
            "type": "string",
            "enum": [
              "Programmatic access",
              "Console access"
            ]
```

```
        },
        "UserPermissions": {
          "description": "Detailed information about the user permissions, or
a policy document to be attached to the user (paste the policy document into the
value field). Details should include the type of access (for example Read, Write or
Delete).",
          "type": "string",
          "minLength": 1,
          "maxLength": 5000
        },
        "Tags": {
          "description": "Up to 50 tags (key/value pairs) to categorize the IAM
User.",
          "type": "array",
          "items": {
            "type": "object",
            "properties": {
              "Key": {
                "type": "string",
                "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                "minLength": 1,
                "maxLength": 127
              },
              "Value": {
                "type": "string",
                "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                "minLength": 1,
                "maxLength": 255
              }
            },
            "additionalProperties": false,
            "metadata": {
              "ui:order": [
                "Key",
                "Value"
              ]
            },
            "required": [
              "Key",
              "Value"
            ]
          },
          "minItems": 0,
          "maxItems": 50,
```

```
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "UserName",
            "AccessType",
            "UserPermissions",
            "Tags"
          ]
        },
        "required": [
          "UserName",
          "AccessType",
          "UserPermissions"
        ]
      },
      "minItems": 0,
      "maxItems": 1
    },
    "IAM Role": {
      "description": "Create IAM role.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "RoleName": {
            "description": "A name for the IAM role. The name can be up to 64
characters in length, and is limited to use characters a-z, A-Z, 0-9, and _+=,.@-.",
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$",
            "minLength": 1,
            "maxLength": 64
          },
          "TrustPolicy": {
            "description": "Detailed information about the trust relationship, or an
assume role policy document to be attached to the role (paste the policy document into
the value field).",
            "type": "string",
            "minLength": 1,
            "maxLength": 5000
          },
          "RolePermissions": {
```

```
            "description": "Detailed information about role permissions, or a policy
document to be attached to the role (paste the policy document into the value field).
Details should include the type of access (for example Read, Write or Delete).",
            "type": "string",
            "minLength": 1,
            "maxLength": 5000
          },
          "Tags": {
            "description": "Up to 50 tags (key/value pairs) to categorize the IAM
role.",
            "type": "array",
            "items": {
              "type": "object",
              "properties": {
                "Key": {
                  "type": "string",
                  "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                  "minLength": 1,
                  "maxLength": 127
                },
                "Value": {
                  "type": "string",
                  "pattern": "^[a-zA-Z0-9\\s_.:/=+@-]+$",
                  "minLength": 1,
                  "maxLength": 255
                }
              },
              "additionalProperties": false,
              "metadata": {
                "ui:order": [
                  "Key",
                  "Value"
                ]
              },
              "required": [
                "Key",
                "Value"
              ]
            },
            "minItems": 0,
            "maxItems": 50,
            "uniqueItems": true
          }
        },
```

```
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "RoleName",
            "TrustPolicy",
            "RolePermissions",
            "Tags"
          ]
        },
        "required": [
          "RoleName",
          "TrustPolicy",
          "RolePermissions"
        ]
      },
      "minItems": 0,
      "maxItems": 1
    },
    "IAM Policy": {
      "description": "Create IAM policy.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "PolicyName": {
            "description": "A name for the IAM policy. The name can be up to 128
characters in length, and is limited to use characters a-z, A-Z, 0-9, and _+=,.@-.",
            "type": "string",
            "pattern": "^[a-zA-Z0-9_+=,.@-]{1,128}$",
            "minLength": 1,
            "maxLength": 64
          },
          "PolicyDocument": {
            "description": "Detailed information about policy permissions, or a policy
document (paste the policy document into the value field).",
            "type": "string",
            "minLength": 1,
            "maxLength": 20480
          },
          "RelatedResources": {
            "description": "IAM users or roles to which the policy applies.",
            "type": "array",
            "items": {
              "type": "string",
```

```
              "minLength": 1,
              "maxLength": 64
            },
            "minItems": 0,
            "maxItems": 10,
            "uniqueItems": true
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "PolicyName",
            "PolicyDocument",
            "RelatedResources"
          ]
        },
        "required": [
          "PolicyName",
          "PolicyDocument",
          "RelatedResources"
        ]
      },
      "minItems": 0,
      "maxItems": 10,
      "uniqueItems": true
    },
    "Operation": {
      "description": "Must be Create.",
      "type": "string",
      "default": "Create",
      "enum": [
        "Create"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
```

```
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "UseCase",
      "IAM User",
      "IAM Role",
      "IAM Policy",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "UseCase",
    "Operation"
  ]
}
```

Schema for Change Type ct-3dscwaeyi6cup

Classifications:

- Deployment | Managed landing zone | Networking account | Create transit gateway route table

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Transit Gateway Route Table",
  "description": "Create a transit gateway (TGW) route table. Use this change type for
 multi-account landing zone (MALZ) Networking accounts only.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateTGWRouteTable.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateTGWRouteTable"
      ],
      "default": "AWSManagedServices-CreateTGWRouteTable"
    },
    "Region": {
      "description": "The AWS region in which the Transit Gateway is located, in the
 form us-east-1.",
```

```
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "TransitGatewayRouteTableName": {
          "description": "The name of the transit gateway route table. Do not
specify these AMS-protected route tables: CoreRouteDomain, DMZBastionsRouteDomain,
EgressRouteDomain, OnPremiseRouteDomain, and defaultAppRouteDomain.",
          "type": "string",
          "pattern": "^[a-zA-Z0-9_+-]{1,256}$"
        },
        "TransitGatewayId": {
          "description": "The ID of the transit gateway, in the form
tgw-01234567891234.",
          "type": "string",
          "pattern": "^tgw-[a-z0-9]{17}$"
        },
        "TGWRouteTableType": {
          "description": "To create an application route table with a static route to
destination: 0.0.0.0/0 going out through the egress VPC attachment, and static routes
to the DMZ VPC and shared services VPC CIDRs, use createApplicationRouteDomain. To
create a custom route table with an empty static route, use createCustomRouteDomain.
The default is createApplicationRouteDomain.",
          "type": "string",
          "default": "createApplicationRouteDomain",
          "enum": [
            "createApplicationRouteDomain",
            "createCustomRouteDomain"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "TransitGatewayRouteTableName",
          "TransitGatewayId",
          "TGWRouteTableType"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayRouteTableName",
        "TransitGatewayId",
```

```
        "TGWRouteTableType"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3e3h8u0sp5z80

Classifications:

- Management | Advanced stack components | EBS Volume | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete EBS Volumes",
  "description": "Delete Elastic Block Store (EBS) volumes in an available state.
 Volumes that are not attached to an instance are in an available state and can be
 deleted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteEBSVolumesV2.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteEBSVolumesV2"
      ],
      "default": "AWSManagedServices-DeleteEBSVolumesV2"
    },
    "Region": {
```

```
      "description": "The AWS Region where the EBS volumes are, in the form us-
east-1.",
      "type": "string",
      "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VolumeIds": {
          "description": "A list of up to 50 EBS volumes to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^vol-[0-9a-f]{8}$|^vol-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 50,
          "uniqueItems": true
        },
        "CreateBackup": {
          "description": "Set to True to create backup snapshots before deleting EBS
 volumes. Leave blank to not create backup snapshots.",
          "type": "boolean",
          "default": true
        },
        "DeleteStackVolume": {
          "description": "Set to True to continue deletion of volume if it is a
 CloudFormation stack resource.",
          "type": "boolean",
          "default": false
        }
      },
      "metadata": {
        "ui:order": [
          "VolumeIds",
          "CreateBackup",
          "DeleteStackVolume"
        ]
      },
      "additionalProperties": false,
      "required": [
        "VolumeIds",
        "CreateBackup",
        "DeleteStackVolume"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Parameters",
      "DocumentName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "Parameters",
    "DocumentName"
  ]
}
```

Schema for Change Type ct-3e3prksxmdhw8

Classifications:

- Deployment | Advanced stack components | AMI | Create from Auto Scaling group

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create AMI From Auto Scaling Group",
  "description": "Create an Amazon Machine Image (AMI) from an EC2 Instance in an Auto
 Scaling group.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateAmiInAutoScalingGroup.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateAmiInAutoScalingGroup"
      ],
      "default": "AWSManagedServices-CreateAmiInAutoScalingGroup"
    },
    "Region": {
      "description": "The AWS Region where the Auto Scaling group is located, in the
 form us-east-1.",
```

```
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AutoScalingGroupName": {
          "description": "The name of the Auto Scaling group to use to create the
AMI.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^.{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Sysprep": {
          "description": "True to Sysprep the Windows instance, False to not. Default
is False. For Linux instances, if set to True, the hostname is reset and the instance
is removed from the domain. If True, ensure that there are at least two EC2 instances
that are in the 'InService' state in the Auto Scaling group. The instance is stopped
and any connected user is logged out from the session. The instance is started after
the AMI is created.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "StopInstance": {
          "description": "True to stop the instance, False to not. Default is False.
If True, ensure that there are at least two EC2 instances that are in the 'InService'
state in the Auto Scaling group. The instance is stopped and any connected user
is logged out from the session. If Sysprep is True, the instance is stopped before
creating the AMI, irrespective of the value you set here. The instance is started
after the AMI is created.",
          "type": "array",
```

```
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "AutoScalingGroupName",
          "Sysprep",
          "StopInstance"
        ]
      },
      "required": [
        "AutoScalingGroupName",
        "Sysprep",
        "StopInstance"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3ebotglihggse

Classifications:

- Deployment | Patching | SSM patch baseline | Create (Red Hat)


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SSM Patch Baseline (Red Hat)",
  "description": "Create an AWS Systems Manager (SSM) patch baseline to define which
 patches are approved for installation on your instances for RHEL OS. Specify existing
 instance \"Patch Group\" tag values for the patch baseline. The patch baseline is an
 SSM resource that you can manage with the SSM console.",
  "additionalProperties": false,
  "properties": {
    "ApprovalRules": {
      "description": "Create auto-approval rules to specify that certain types of
 operating system patches are approved automatically.",
      "items": {
        "additionalProperties": false,
        "properties": {
          "ApproveAfterDays": {
            "default": 7,
            "description": "The number of days to wait after a patch is released before
 approving patches automatically.",
            "maximum": 100,
            "minimum": 0,
            "type": "integer"
          },
          "Classification": {
            "description": "The Classification of the patches to be selected. Allowed
 values are \"All\", \"Bugfix\", \"Enhancement\", \"Newpackage\", \"Recommended\" and
 \"Security\".",
            "items": {
              "enum": [
                "All",
                "Bugfix",
                "Enhancement",
                "Newpackage",
                "Recommended",
                "Security"
              ],
```

```
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          },
          "Severity": {
            "description": "The severity of the patches to be selected. Allowed values
 are \"All\", \"Critical\", \"Important\", \"Low\", \"Moderate\" and \"None\".",
            "items": {
              "enum": [
                "All",
                "Critical",
                "Important",
                "Low",
                "Moderate",
                "None"
              ],
              "type": "string"
            },
            "type": "array",
            "uniqueItems": true
          }
        },
        "metadata": {
          "ui:order": [
            "Severity",
            "Classification",
            "ApproveAfterDays"
          ]
        },
        "required": [
          "ApproveAfterDays"
        ],
        "type": "object"
      },
      "maxItems": 10,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "ApprovedPatches": {
      "description": "The list of patches to approve explicitly.",
      "items": {
        "type": "string",
```

```
        "maxLength": 100,
        "minLength": 1
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Description": {
      "description": "A meaningful description for this patch baseline.",
      "maxLength": 500,
      "minLength": 1,
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this patch baseline.",
      "maxLength": 128,
      "minLength": 3,
      "pattern": "^[a-zA-Z0-9._-]+$",
      "type": "string"
    },
    "OperatingSystem": {
      "default": "Red Hat Enterprise Linux",
      "description": "The operating system of instances to which this baseline is
applied.",
      "enum": [
        "Red Hat Enterprise Linux"
      ],
      "type": "string"
    },
    "PatchGroupTagValues": {
      "description": "A list of the values of your \"Patch Group\" tags on the
instances you want patched; the values for up to twenty-five \"Patch Group\" tags can
be provided. Instances with those values are associated with this patch baseline.",
      "items": {
        "maxLength": 256,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 25,
      "minItems": 1,
      "type": "array",
      "uniqueItems": true
    },
```

```
    "RejectedPatches": {
      "description": "The list of patches to reject explicitly.",
      "items": {
        "maxLength": 100,
        "minLength": 1,
        "type": "string"
      },
      "maxItems": 50,
      "minItems": 0,
      "type": "array",
      "uniqueItems": true
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the SSM patch
 baseline resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
```

```
      "uniqueItems": true
    }
  },
  "metadata": {
    "ui:order": [
      "OperatingSystem",
      "Name",
      "Description",
      "PatchGroupTagValues",
      "ApprovalRules",
      "ApprovedPatches",
      "RejectedPatches",
      "Tags"
    ]
  },
  "required": [
    "Name",
    "PatchGroupTagValues",
    "OperatingSystem"
  ],
  "type": "object"
}
```

Schema for Change Type ct-3eutt7grkict4

Classifications:

- Management | Directory Service | Users and groups | Add group

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add AD Group",
  "description": "Create an Active Directory (AD) group in the AMS managed AD. For
 multi-account landing zone (MALZ), use this change type in the shared services
 account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateADGroup-Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateADGroup-Admin"
```

```
      ],
      "default": "AWSManagedServices-CreateADGroup-Admin"
    },
    "Region": {
      "description": "The AWS Region where the AMS managed AD is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "GroupName": {
          "description": "A meaningful name for the AD group. It must contain 2 to 63
 characters and cannot contain the following special characters: /\\[]:;|=,+*?<>\" or a
 leading or trailing space.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(?!\\.+$)(?!\\d+$)(?! +$)[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|
\\\\][^#,\\+\"\\<>;\r\n\f\\[\\]\\*:=?/\\|\\\\]{0,61}[^ #,\\+\"\\<>;\r\n\f\\[\\]\\*:=/\
\|]$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "GroupDescription": {
          "description": "A description for the new group.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^.{1,1024}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "GroupScope": {
          "description": "The scope for the new group. Default is DomainLocal. For
 current definitions see Microsoft AD documentation.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "DomainLocal",
```

```
              "Global",
              "Universal"
            ],
            "default": "DomainLocal"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "GroupName",
          "GroupDescription",
          "GroupScope"
        ]
      },
      "required": [
        "GroupName",
        "GroupDescription"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3fi2cx8b83iua

Classifications:

- Management | Advanced stack components | Auto scaling group | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update an Auto Scaling Group",
  "description": "Update an Auto Scaling Group and associated launch configuration
 created with CT ct-2tylseo8rxfsc, version 2.0.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC that contains the Auto Scaling Group in the form
 vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "StackId": {
      "description": "The stack ID of the Auto Scaling Group that you are updating, in
 the form stack-a1b2c3d4e5f67890e.",
      "type": "string",
      "pattern": "^stack-[a-z0-9]{17}$"
    },
    "Parameters": {
      "description": "Specifications for updating the Auto Scaling Group.",
      "type": "object",
      "properties": {
        "ASGAmiId": {
          "description": "The AMI for the Auto Scaling Group update. All instances in
 the group are replaced if this is updated.",
          "type": "string",
          "pattern": "^ami-[a-z0-9]{8}$|^ami-[a-z0-9]{17}$"
        },
        "ASGCooldown": {
          "description": "The number of seconds after a scaling activity is complete
 before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600
        },
        "ASGDesiredCapacity": {
          "description": "The number of EC2 instances you want running in the group.
 This number must be greater than or equal to the ASGMinInstances setting and less than
 or equal to the ASGMaxInstances setting.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000
```

```
      },
      "ASGEBSOptimized": {
        "description": "True to create EBS-optimized instances, false to not. All
instances in the group are replaced if this is updated.",
        "type": "string",
        "enum": [
          "true",
          "false"
        ]
      },
      "ASGHealthCheckGracePeriod": {
        "description": "The amount of time, in seconds, that Auto Scaling waits
before checking the health status of an EC2 instance that has come into service.
During this time, any health check failures for the instance are ignored.",
        "type": "integer",
        "minimum": 600,
        "maximum": 1800
      },
      "ASGHealthCheckType": {
        "description": "The service to use for the health checks. The ELB
Health Check Type includes EC2 instance and system status checks. Only choose
ELB as the ASGHealthCheckType if the ASG is being fronted by Load Balancers. If
ASGHealthCheckType = ELB, ensure that your ASGHealthCheckGracePeriod value is long
enough so that your instances are not terminated due to load-balancer health checks
failing, before your application has been deployed.",
        "type": "string",
        "enum": [
          "EC2",
          "ELB"
        ]
      },
      "ASGIAMInstanceProfile": {
        "description": "The IAM instance profile name for the Auto Scaling group.
EC2 instances launched with an IAM role automatically have AWS security credentials
available. All instances in the group are replaced if this is updated.",
        "type": "string",
        "pattern": "^customer[\\w-]+$"
      },
      "ASGInstanceDetailedMonitoring": {
        "description": "True to enable detailed monitoring on the instances in the
Auto Scaling group, false to use only basic monitoring. All instances in the group are
replaced if this is updated.",
        "type": "string",
        "enum": [
```

```
            "true",
            "false"
          ]
        },
        "ASGInstanceRootVolumeIops": {
          "description": "The Iops to use for the root volume if
ASGInstanceRootVolumeType = io1. All instances in the group are replaced if this is
updated.",
          "type": "integer",
          "minimum": 0,
          "maximum": 20000
        },
        "ASGInstanceRootVolumeSize": {
          "description": "The size of the root volume for the instance. All instances
in the group are replaced if this is updated.",
          "type": "integer",
          "minimum": 8,
          "maximum": 16000
        },
        "ASGInstanceRootVolumeType": {
          "description": "Choose io1 or gp2 for SSD-backed volumes optimized for
transactional workloads; choose standard for HDD-backed volumes optimized for large
streaming workloads. All instances in the group are replaced if this is updated.",
          "type": "string",
          "enum": [
            "standard",
            "io1",
            "gp2"
          ]
        },
        "ASGInstanceType": {
          "description": "The instance type for the Auto Scaling group instances to
update to. All instances in the group are replaced if this is updated.",
          "type": "string"
        },
        "ASGLoadBalancerNames": {
          "description": "A list of load balancers to associate with this Auto Scaling
group. Use Classic Load Balancer (ELBs).",
          "type": "array",
          "items": {
            "type": "string"
          },
          "minItems": 1,
          "maxItems": 10,
```

```
          "uniqueItems": true
        },
        "ASGMaxInstances": {
          "description": "The maximum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000
        },
        "ASGMinInstances": {
          "description": "The minimum number of instances you want in the Auto Scaling
group at any time.",
          "type": "integer",
          "minimum": 1,
          "maximum": 1000
        },
        "ASGScaleDownMetricName": {
          "description": "The metric to use to in a scale-down event. Exceeding the
metric triggers an alarm.",
          "type": "string",
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
            "StatusCheckFailed_System"
          ]
        },
        "ASGScaleDownPolicyCooldown": {
          "description": "The number of seconds after a scale-down activity is
completed before any further scaling activities can start.",
          "type": "integer",
          "minimum": 120,
          "maximum": 600
        },
        "ASGScaleDownPolicyEvaluationPeriods": {
```

```
          "description": "The number of periods over which data is compared to the
specified ASGScaleDownMetricName threshold.",
          "type": "integer",
          "minimum": 2
        },
        "ASGScaleDownPolicyPeriod": {
          "description": "The time over which the specified ASGScaleDownPolicyStatistic
is applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60
        },
        "ASGScaleDownPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale down.",
          "type": "integer",
          "maximum": 0
        },
        "ASGScaleDownPolicyStatistic": {
          "description": "The statistic to apply to the alarm's
ASGScaleDownMetricName.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ]
        },
        "ASGScaleDownPolicyThreshold": {
          "description": "The value against which the specified
ASGScaleDownPolicyStatistic is compared.",
          "type": "number"
        },
        "ASGScaleUpMetricName": {
          "description": "The metric to use in a scale-up event. Exceeding the metric
triggers an alarm.",
          "type": "string",
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
```

```
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
            "StatusCheckFailed_System"
          ]
        },
        "ASGScaleUpPolicyCooldown": {
          "description": "The amount of time, in seconds, after a scale-up activity is
completed before any further trigger-related scaling activities can start.",
          "type": "integer",
          "minimum": 60
        },
        "ASGScaleUpPolicyEvaluationPeriods": {
          "description": "The number of periods over which data is compared to the
specified ASGScaleUpMetricName threshold.",
          "type": "integer",
          "minimum": 2
        },
        "ASGScaleUpPolicyPeriod": {
          "description": "The time over which the specified ASGScaleUpPolicyStatistic
is applied. You must specify a time in seconds that is a multiple of 60.",
          "type": "integer",
          "multipleOf": 60,
          "minimum": 60
        },
        "ASGScaleUpPolicyScalingAdjustment": {
          "description": "The number of instances by which to scale up.",
          "type": "integer",
          "minimum": 0
        },
        "ASGScaleUpPolicyStatistic": {
          "description": "The statistic to apply to the alarm's
ASGScaleUpMetricName.",
          "type": "string",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ]
```

```
        },
        "ASGScaleUpPolicyThreshold": {
          "description": "The value against which the specified
ASGScaleUpPolicyStatistic is compared.",
          "type": "number"
        },
        "ASGSubnetIds": {
          "description": "One or more subnets for the Auto Scaling group to launch
instances into (scale up) or remove instances from (scale down), in the form
subnet-12345678. All instances in the group are replaced if this is updated.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 2,
          "uniqueItems": true
        },
        "ASGUserData": {
          "description": "A newline delimited list where each element is a line
of script to be run on boot. All instances in the group are replaced if this is
updated.",
          "type": "string"
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "ASGAmiId",
          "ASGCooldown",
          "ASGDesiredCapacity",
          "ASGEBSOptimized",
          "ASGHealthCheckGracePeriod",
          "ASGHealthCheckType",
          "ASGIAMInstanceProfile",
          "ASGInstanceDetailedMonitoring",
          "ASGInstanceRootVolumeIops",
          "ASGInstanceRootVolumeSize",
          "ASGInstanceRootVolumeType",
          "ASGInstanceType",
          "ASGLoadBalancerNames",
          "ASGMaxInstances",
          "ASGMinInstances",
```

```
          "ASGScaleDownMetricName",
          "ASGScaleDownPolicyCooldown",
          "ASGScaleDownPolicyEvaluationPeriods",
          "ASGScaleDownPolicyPeriod",
          "ASGScaleDownPolicyScalingAdjustment",
          "ASGScaleDownPolicyStatistic",
          "ASGScaleDownPolicyThreshold",
          "ASGScaleUpMetricName",
          "ASGScaleUpPolicyCooldown",
          "ASGScaleUpPolicyEvaluationPeriods",
          "ASGScaleUpPolicyPeriod",
          "ASGScaleUpPolicyScalingAdjustment",
          "ASGScaleUpPolicyStatistic",
          "ASGScaleUpPolicyThreshold",
          "ASGSubnetIds",
          "ASGUserData"
        ]
      }
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackId",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "StackId",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3g6fq83nxg1a7

Classifications:

- Management | Advanced stack components | Application Load Balancer | Add listener certificate

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add ALB Listener Certificate",
  "description": "Add a certificate to the specified Application Load Balancer (ALB)
 listener. Use the RemediateStackDrift parameter for the automation to try to remediate
 drift, if it is introduced.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddCertificateToElbv2Listener.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddCertificateToElbv2Listener"
      ],
      "default": "AWSManagedServices-AddCertificateToElbv2Listener"
    },
    "Region": {
      "description": "The AWS Region where the application load balancer listener is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ListenerArn": {
          "description": "The Amazon Resource Name (ARN) of the listener in
 the form arn:aws:elasticloadbalancing:us-east-1:123456789012:listener/app/
sample/1234567890abcdfe/1234567890abcdfe.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):elasticloadbalancing:[a-z]{2}-[a-
z]+-[0-9]{1}:[0-9]{12}:listener/[a-z]{3}/[A-Za-z0-9-]+/[a-z0-9-]+/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "CertificateArn": {
          "description": "The Amazon Resource Name (ARN) of the certificate in the form
 arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "type": "array",
          "items": {
            "type": "string",
```

```
            "pattern": "^arn:(aws|aws-cn|aws-us-gov):acm:[a-z]{2}-[a-z]+-[0-9]{1}:[0-9]
{12}:certificate/[a-z0-9-]+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IsDefault": {
          "description": "True to set the certificate as the default certificate on the
 listener, False to not set the certificate as the default certificate on the listener.
 Default value is False.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "False",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        },
        "RemediateStackDrift": {
          "description": "True to initiate drift remediation, if any drift is caused
 by adding the certificate to the Load Balancer listener. False to not attempt drift
 remediation. Drift remediation can be performed only on CloudFormation stacks that
 were created using a CT other than the Ingestion CT ct-36cn2avfrrj9v and that are
 in sync with the definitions in the stack template prior to adding certificate to
 the Load Balancer listener. Set to False to add the certificate to the Load Balancer
 listener in an ingested stack if any drift introduced by the change is acceptable.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "True",
            "enum": [
              "True",
              "False"
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
```

```
        "ui:order": [
          "ListenerArn",
          "CertificateArn",
          "IsDefault",
          "RemediateStackDrift"
        ]
      },
      "additionalProperties": false,
      "required": [
        "CertificateArn",
        "ListenerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3g9dbtun44mal

Classifications:

- Management | Advanced stack components | EC2 instance stack | Change time zone

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Change Timezone",
  "description": "Change the time zone of an EC2 instance. To reboot the EC2 instance
 after changing the time zone, set Reboot = true.",
  "type": "object",
  "properties": {
```

```
    "DocumentName": {
      "description": "Must be AWSManagedServices-SetInstanceTimeZone.",
      "type": "string",
      "enum": [
        "AWSManagedServices-SetInstanceTimeZone"
      ],
      "default": "AWSManagedServices-SetInstanceTimeZone"
    },
    "Region": {
      "description": "The AWS Region in which the resource is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "The ID of the instance, in the form i-12345678901234567 or
 i-12345678.",
          "type": "string",
          "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
        },
        "Reboot": {
          "description": "True to reboot the EC2 instance after changing the time zone.
 False to not reboot.",
          "type": "string",
          "default": "False",
          "enum": [
            "True",
            "False"
          ]
        },
        "TimeZone": {
          "description": "The time zone to set on the EC2 instance, in the form
 Australia/Sydney (AUS Eastern Standard Time).",
          "type": "string",
          "enum": [
            "Africa/Abidjan (Greenwich Standard Time)",
            "Africa/Accra (Greenwich Standard Time)",
            "Africa/Addis_Ababa (E. Africa Standard Time)",
            "Africa/Algiers (W. Central Africa Standard Time)",
            "Africa/Bamako (Greenwich Standard Time)",
            "Africa/Bangui (W. Central Africa Standard Time)",
```

```
            "Africa/Banjul (Greenwich Standard Time)",
            "Africa/Bissau (Greenwich Standard Time)",
            "Africa/Blantyre (South Africa Standard Time)",
            "Africa/Brazzaville (W. Central Africa Standard Time)",
            "Africa/Bujumbura (South Africa Standard Time)",
            "Africa/Cairo (Egypt Standard Time)",
            "Africa/Casablanca (Morocco Standard Time)",
            "Africa/Ceuta (Romance Standard Time)",
            "Africa/Conakry (Greenwich Standard Time)",
            "Africa/Dakar (Greenwich Standard Time)",
            "Africa/Dar_es_Salaam (E. Africa Standard Time)",
            "Africa/Djibouti (E. Africa Standard Time)",
            "Africa/Douala (W. Central Africa Standard Time)",
            "Africa/El_Aaiun (Morocco Standard Time)",
            "Africa/Freetown (Greenwich Standard Time)",
            "Africa/Gaborone (South Africa Standard Time)",
            "Africa/Harare (South Africa Standard Time)",
            "Africa/Johannesburg (South Africa Standard Time)",
            "Africa/Juba (E. Africa Standard Time)",
            "Africa/Kampala (E. Africa Standard Time)",
            "Africa/Khartoum (Sudan Standard Time)",
            "Africa/Kigali (South Africa Standard Time)",
            "Africa/Kinshasa (W. Central Africa Standard Time)",
            "Africa/Lagos (W. Central Africa Standard Time)",
            "Africa/Libreville (W. Central Africa Standard Time)",
            "Africa/Lome (Greenwich Standard Time)",
            "Africa/Luanda (W. Central Africa Standard Time)",
            "Africa/Lubumbashi (South Africa Standard Time)",
            "Africa/Lusaka (South Africa Standard Time)",
            "Africa/Malabo (W. Central Africa Standard Time)",
            "Africa/Maputo (South Africa Standard Time)",
            "Africa/Maseru (South Africa Standard Time)",
            "Africa/Mbabane (South Africa Standard Time)",
            "Africa/Mogadishu (E. Africa Standard Time)",
            "Africa/Monrovia (Greenwich Standard Time)",
            "Africa/Nairobi (E. Africa Standard Time)",
            "Africa/Ndjamena (W. Central Africa Standard Time)",
            "Africa/Niamey (W. Central Africa Standard Time)",
            "Africa/Nouakchott (Greenwich Standard Time)",
            "Africa/Ouagadougou (Greenwich Standard Time)",
            "Africa/Porto-Novo (W. Central Africa Standard Time)",
            "Africa/Sao_Tome (Sao Tome Standard Time)",
            "Africa/Tripoli (Libya Standard Time)",
            "Africa/Tunis (W. Central Africa Standard Time)",
```

```
            "Africa/Windhoek (Namibia Standard Time)",
            "America/Adak (Aleutian Standard Time)",
            "America/Anchorage (Alaskan Standard Time)",
            "America/Anguilla (SA Western Standard Time)",
            "America/Antigua (SA Western Standard Time)",
            "America/Araguaina (Tocantins Standard Time)",
            "America/Argentina/La_Rioja (Argentina Standard Time)",
            "America/Argentina/Rio_Gallegos (Argentina Standard Time)",
            "America/Argentina/Salta (Argentina Standard Time)",
            "America/Argentina/San_Juan (Argentina Standard Time)",
            "America/Argentina/San_Luis (Argentina Standard Time)",
            "America/Argentina/Tucuman (Argentina Standard Time)",
            "America/Argentina/Ushuaia (Argentina Standard Time)",
            "America/Aruba (SA Western Standard Time)",
            "America/Asuncion (Paraguay Standard Time)",
            "America/Bahia (Bahia Standard Time)",
            "America/Bahia_Banderas (Central Standard Time (Mexico))",
            "America/Barbados (SA Western Standard Time)",
            "America/Belem (SA Eastern Standard Time)",
            "America/Belize (Central America Standard Time)",
            "America/Blanc-Sablon (SA Western Standard Time)",
            "America/Boa_Vista (SA Western Standard Time)",
            "America/Bogota (SA Pacific Standard Time)",
            "America/Boise (Mountain Standard Time)",
            "America/Buenos_Aires (Argentina Standard Time)",
            "America/Cambridge_Bay (Mountain Standard Time)",
            "America/Campo_Grande (Central Brazilian Standard Time)",
            "America/Cancun (Eastern Standard Time (Mexico))",
            "America/Caracas (Venezuela Standard Time)",
            "America/Cayenne (SA Eastern Standard Time)",
            "America/Cayman (SA Pacific Standard Time)",
            "America/Chicago (Central Standard Time)",
            "America/Chihuahua (Mountain Standard Time (Mexico))",
            "America/Costa_Rica (Central America Standard Time)",
            "America/Creston (US Mountain Standard Time)",
            "America/Cuiaba (Central Brazilian Standard Time)",
            "America/Curacao (SA Western Standard Time)",
            "America/Danmarkshavn (UTC)",
            "America/Dawson (Pacific Standard Time)",
            "America/Dawson_Creek (US Mountain Standard Time)",
            "America/Denver (Mountain Standard Time)",
            "America/Detroit (Eastern Standard Time)",
            "America/Dominica (SA Western Standard Time)",
            "America/Edmonton (Mountain Standard Time)",
```

```
            "America/Eirunepe (SA Pacific Standard Time)",
            "America/El_Salvador (Central America Standard Time)",
            "America/Fortaleza (SA Eastern Standard Time)",
            "America/Glace_Bay (Atlantic Standard Time)",
            "America/Godthab (Greenland Standard Time)",
            "America/Goose_Bay (Atlantic Standard Time)",
            "America/Grand_Turk (Turks And Caicos Standard Time)",
            "America/Grenada (SA Western Standard Time)",
            "America/Guadeloupe (SA Western Standard Time)",
            "America/Guatemala (Central America Standard Time)",
            "America/Guayaquil (SA Pacific Standard Time)",
            "America/Guyana (SA Western Standard Time)",
            "America/Halifax (Atlantic Standard Time)",
            "America/Havana (Cuba Standard Time)",
            "America/Hermosillo (US Mountain Standard Time)",
            "America/Indiana/Knox (Central Standard Time)",
            "America/Indiana/Marengo (US Eastern Standard Time)",
            "America/Indiana/Petersburg (Eastern Standard Time)",
            "America/Indiana/Tell_City (Central Standard Time)",
            "America/Indiana/Vevay (US Eastern Standard Time)",
            "America/Indiana/Vincennes (Eastern Standard Time)",
            "America/Indiana/Winamac (Eastern Standard Time)",
            "America/Indianapolis (US Eastern Standard Time)",
            "America/Inuvik (Mountain Standard Time)",
            "America/Iqaluit (Eastern Standard Time)",
            "America/Jamaica (SA Pacific Standard Time)",
            "America/Juneau (Alaskan Standard Time)",
            "America/Kentucky/Monticello (Eastern Standard Time)",
            "America/Kralendijk (SA Western Standard Time)",
            "America/La_Paz (SA Western Standard Time)",
            "America/Lima (SA Pacific Standard Time)",
            "America/Los_Angeles (Pacific Standard Time)",
            "America/Lower_Princes (SA Western Standard Time)",
            "America/Maceio (SA Eastern Standard Time)",
            "America/Managua (Central America Standard Time)",
            "America/Manaus (SA Western Standard Time)",
            "America/Marigot (SA Western Standard Time)",
            "America/Martinique (SA Western Standard Time)",
            "America/Matamoros (Central Standard Time)",
            "America/Mazatlan (Mountain Standard Time (Mexico))",
            "America/Menominee (Central Standard Time)",
            "America/Merida (Central Standard Time (Mexico))",
            "America/Metlakatla (Alaskan Standard Time)",
            "America/Mexico_City (Central Standard Time (Mexico))",
```

```
            "America/Miquelon (Saint Pierre Standard Time)",
            "America/Moncton (Atlantic Standard Time)",
            "America/Monterrey (Central Standard Time (Mexico))",
            "America/Montevideo (Montevideo Standard Time)",
            "America/Montreal (Eastern Standard Time)",
            "America/Montserrat (SA Western Standard Time)",
            "America/Nassau (Eastern Standard Time)",
            "America/New_York (Eastern Standard Time)",
            "America/Nipigon (Eastern Standard Time)",
            "America/Nome (Alaskan Standard Time)",
            "America/Noronha (UTC-02)",
            "America/North_Dakota/Beulah (Central Standard Time)",
            "America/North_Dakota/Center (Central Standard Time)",
            "America/North_Dakota/New_Salem (Central Standard Time)",
            "America/Ojinaga (Mountain Standard Time)",
            "America/Panama (SA Pacific Standard Time)",
            "America/Pangnirtung (Eastern Standard Time)",
            "America/Paramaribo (SA Eastern Standard Time)",
            "America/Phoenix (US Mountain Standard Time)",
            "America/Port-au-Prince (Haiti Standard Time)",
            "America/Port_of_Spain (SA Western Standard Time)",
            "America/Porto_Velho (SA Western Standard Time)",
            "America/Puerto_Rico (SA Western Standard Time)",
            "America/Punta_Arenas (Magallanes Standard Time)",
            "America/Rainy_River (Central Standard Time)",
            "America/Rankin_Inlet (Central Standard Time)",
            "America/Recife (SA Eastern Standard Time)",
            "America/Regina (Canada Central Standard Time)",
            "America/Resolute (Central Standard Time)",
            "America/Rio_Branco (SA Pacific Standard Time)",
            "America/Santa_Isabel (Pacific Standard Time (Mexico))",
            "America/Santarem (SA Eastern Standard Time)",
            "America/Santiago (Pacific SA Standard Time)",
            "America/Santo_Domingo (SA Western Standard Time)",
            "America/Sao_Paulo (E. South America Standard Time)",
            "America/Scoresbysund (Azores Standard Time)",
            "America/Sitka (Alaskan Standard Time)",
            "America/St_Barthelemy (SA Western Standard Time)",
            "America/St_Johns (Newfoundland Standard Time)",
            "America/St_Kitts (SA Western Standard Time)",
            "America/St_Lucia (SA Western Standard Time)",
            "America/St_Thomas (SA Western Standard Time)",
            "America/St_Vincent (SA Western Standard Time)",
            "America/Swift_Current (Canada Central Standard Time)",
```

```
            "America/Tegucigalpa (Central America Standard Time)",
            "America/Thule (Atlantic Standard Time)",
            "America/Thunder_Bay (Eastern Standard Time)",
            "America/Tijuana (Pacific Standard Time (Mexico))",
            "America/Toronto (Eastern Standard Time)",
            "America/Tortola (SA Western Standard Time)",
            "America/Vancouver (Pacific Standard Time)",
            "America/Whitehorse (Pacific Standard Time)",
            "America/Winnipeg (Central Standard Time)",
            "America/Yakutat (Alaskan Standard Time)",
            "America/Yellowknife (Mountain Standard Time)",
            "Antarctica/Casey (Singapore Standard Time)",
            "Antarctica/Davis (SE Asia Standard Time)",
            "Antarctica/DumontDUrville (West Pacific Standard Time)",
            "Antarctica/Macquarie (Central Pacific Standard Time)",
            "Antarctica/Mawson (West Asia Standard Time)",
            "Antarctica/McMurdo (New Zealand Standard Time)",
            "Antarctica/Palmer (SA Eastern Standard Time)",
            "Antarctica/Rothera (SA Eastern Standard Time)",
            "Antarctica/Syowa (E. Africa Standard Time)",
            "Antarctica/Vostok (Central Asia Standard Time)",
            "Arctic/Longyearbyen (W. Europe Standard Time)",
            "Asia/Aden (Arab Standard Time)",
            "Asia/Almaty (Central Asia Standard Time)",
            "Asia/Amman (Jordan Standard Time)",
            "Asia/Anadyr (Russia Time Zone 11)",
            "Asia/Aqtau (West Asia Standard Time)",
            "Asia/Aqtobe (West Asia Standard Time)",
            "Asia/Ashgabat (West Asia Standard Time)",
            "Asia/Baghdad (Arabic Standard Time)",
            "Asia/Bahrain (Arab Standard Time)",
            "Asia/Baku (Azerbaijan Standard Time)",
            "Asia/Bangkok (SE Asia Standard Time)",
            "Asia/Barnaul (Altai Standard Time)",
            "Asia/Beirut (Middle East Standard Time)",
            "Asia/Bishkek (Central Asia Standard Time)",
            "Asia/Brunei (Singapore Standard Time)",
            "Asia/Calcutta (India Standard Time)",
            "Asia/Chita (Transbaikal Standard Time)",
            "Asia/Choibalsan (Ulaanbaatar Standard Time)",
            "Asia/Chongqing (China Standard Time)",
            "Asia/Colombo (Sri Lanka Standard Time)",
            "Asia/Damascus (Syria Standard Time)",
            "Asia/Dhaka (Bangladesh Standard Time)",
```

```
            "Asia/Dili (Tokyo Standard Time)",
            "Asia/Dubai (Arabian Standard Time)",
            "Asia/Dushanbe (West Asia Standard Time)",
            "Asia/Gaza (West Bank Standard Time)",
            "Asia/Harbin (China Standard Time)",
            "Asia/Hebron (West Bank Standard Time)",
            "Asia/Hong_Kong (China Standard Time)",
            "Asia/Hovd (W. Mongolia Standard Time)",
            "Asia/Irkutsk (North Asia East Standard Time)",
            "Asia/Jakarta (SE Asia Standard Time)",
            "Asia/Jayapura (Tokyo Standard Time)",
            "Asia/Jerusalem (Israel Standard Time)",
            "Asia/Kabul (Afghanistan Standard Time)",
            "Asia/Kamchatka (Russia Time Zone 11)",
            "Asia/Karachi (Pakistan Standard Time)",
            "Asia/Kashgar (Central Asia Standard Time)",
            "Asia/Katmandu (Nepal Standard Time)",
            "Asia/Khandyga (Yakutsk Standard Time)",
            "Asia/Krasnoyarsk (North Asia Standard Time)",
            "Asia/Kuala_Lumpur (Singapore Standard Time)",
            "Asia/Kuching (Singapore Standard Time)",
            "Asia/Kuwait (Arab Standard Time)",
            "Asia/Macau (China Standard Time)",
            "Asia/Magadan (Magadan Standard Time)",
            "Asia/Makassar (Singapore Standard Time)",
            "Asia/Manila (Singapore Standard Time)",
            "Asia/Muscat (Arabian Standard Time)",
            "Asia/Nicosia (GTB Standard Time)",
            "Asia/Novokuznetsk (North Asia Standard Time)",
            "Asia/Novosibirsk (N. Central Asia Standard Time)",
            "Asia/Omsk (Omsk Standard Time)",
            "Asia/Oral (West Asia Standard Time)",
            "Asia/Phnom_Penh (SE Asia Standard Time)",
            "Asia/Pontianak (SE Asia Standard Time)",
            "Asia/Pyongyang (North Korea Standard Time)",
            "Asia/Qatar (Arab Standard Time)",
            "Asia/Qyzylorda (Qyzylorda Standard Time)",
            "Asia/Rangoon (Myanmar Standard Time)",
            "Asia/Riyadh (Arab Standard Time)",
            "Asia/Sakhalin (Sakhalin Standard Time)",
            "Asia/Samarkand (West Asia Standard Time)",
            "Asia/Seoul (Korea Standard Time)",
            "Asia/Shanghai (China Standard Time)",
            "Asia/Singapore (Singapore Standard Time)",
```

```
            "Asia/Srednekolymsk (Russia Time Zone 10)",
            "Asia/Taipei (Taipei Standard Time)",
            "Asia/Tashkent (West Asia Standard Time)",
            "Asia/Tbilisi (Georgian Standard Time)",
            "Asia/Tehran (Iran Standard Time)",
            "Asia/Thimphu (Bangladesh Standard Time)",
            "Asia/Tokyo (Tokyo Standard Time)",
            "Asia/Tomsk (Tomsk Standard Time)",
            "Asia/Ulaanbaatar (Ulaanbaatar Standard Time)",
            "Asia/Urumqi (Central Asia Standard Time)",
            "Asia/Ust-Nera (Vladivostok Standard Time)",
            "Asia/Vientiane (SE Asia Standard Time)",
            "Asia/Vladivostok (Vladivostok Standard Time)",
            "Asia/Yakutsk (Yakutsk Standard Time)",
            "Asia/Yekaterinburg (Ekaterinburg Standard Time)",
            "Asia/Yerevan (Caucasus Standard Time)",
            "Atlantic/Azores (Azores Standard Time)",
            "Atlantic/Bermuda (Atlantic Standard Time)",
            "Atlantic/Canary (GMT Standard Time)",
            "Atlantic/Cape_Verde (Cape Verde Standard Time)",
            "Atlantic/Madeira (GMT Standard Time)",
            "Atlantic/Reykjavik (Greenwich Standard Time)",
            "Atlantic/South_Georgia (UTC-02)",
            "Atlantic/St_Helena (Greenwich Standard Time)",
            "Atlantic/Stanley (SA Eastern Standard Time)",
            "Australia/Adelaide (Cen. Australia Standard Time)",
            "Australia/Brisbane (E. Australia Standard Time)",
            "Australia/Broken_Hill (Cen. Australia Standard Time)",
            "Australia/Currie (Tasmania Standard Time)",
            "Australia/Darwin (AUS Central Standard Time)",
            "Australia/Eucla (Aus Central W. Standard Time)",
            "Australia/Hobart (Tasmania Standard Time)",
            "Australia/Lindeman (E. Australia Standard Time)",
            "Australia/Lord_Howe (Lord Howe Standard Time)",
            "Australia/Melbourne (AUS Eastern Standard Time)",
            "Australia/Perth (W. Australia Standard Time)",
            "Australia/Sydney (AUS Eastern Standard Time)",
            "Canada/Atlantic (Atlantic Standard Time)",
            "Canada/Central (Central Standard Time)",
            "Canada/Eastern (Eastern Standard Time)",
            "Canada/Mountain (Mountain Standard Time)",
            "Canada/Newfoundland (Newfoundland Standard Time)",
            "Canada/Pacific (Pacific Standard Time)",
            "Etc/GMT (UTC)",
```

```
            "Etc/GMT+11 (UTC-11)",
            "Etc/GMT+12 (Dateline Standard Time)",
            "Etc/GMT+2 (UTC-02)",
            "Etc/GMT+8 (UTC-08)",
            "Etc/GMT+9 (UTC-09)",
            "Etc/GMT-12 (UTC+12)",
            "Etc/GMT-13 (UTC+13)",
            "Europe/Amsterdam (W. Europe Standard Time)",
            "Europe/Andorra (W. Europe Standard Time)",
            "Europe/Astrakhan (Astrakhan Standard Time)",
            "Europe/Athens (GTB Standard Time)",
            "Europe/Belgrade (Central Europe Standard Time)",
            "Europe/Berlin (W. Europe Standard Time)",
            "Europe/Bratislava (Central Europe Standard Time)",
            "Europe/Brussels (Romance Standard Time)",
            "Europe/Bucharest (GTB Standard Time)",
            "Europe/Budapest (Central Europe Standard Time)",
            "Europe/Busingen (W. Europe Standard Time)",
            "Europe/Chisinau (E. Europe Standard Time)",
            "Europe/Copenhagen (Romance Standard Time)",
            "Europe/Dublin (GMT Standard Time)",
            "Europe/Gibraltar (W. Europe Standard Time)",
            "Europe/Guernsey (GMT Standard Time)",
            "Europe/Helsinki (FLE Standard Time)",
            "Europe/Isle_of_Man (GMT Standard Time)",
            "Europe/Istanbul (Turkey Standard Time)",
            "Europe/Jersey (GMT Standard Time)",
            "Europe/Kaliningrad (Kaliningrad Standard Time)",
            "Europe/Kiev (FLE Standard Time)",
            "Europe/Lisbon (GMT Standard Time)",
            "Europe/Ljubljana (Central Europe Standard Time)",
            "Europe/London (GMT Standard Time)",
            "Europe/Luxembourg (W. Europe Standard Time)",
            "Europe/Madrid (Romance Standard Time)",
            "Europe/Malta (W. Europe Standard Time)",
            "Europe/Mariehamn (FLE Standard Time)",
            "Europe/Minsk (Belarus Standard Time)",
            "Europe/Monaco (W. Europe Standard Time)",
            "Europe/Moscow (Russian Standard Time)",
            "Europe/Oslo (W. Europe Standard Time)",
            "Europe/Paris (Romance Standard Time)",
            "Europe/Podgorica (Central Europe Standard Time)",
            "Europe/Prague (Central Europe Standard Time)",
            "Europe/Riga (FLE Standard Time)",
```

```
            "Europe/Rome (W. Europe Standard Time)",
            "Europe/Samara (Russia Time Zone 3)",
            "Europe/San_Marino (W. Europe Standard Time)",
            "Europe/Sarajevo (Central European Standard Time)",
            "Europe/Saratov (Saratov Standard Time)",
            "Europe/Simferopol (Russian Standard Time)",
            "Europe/Skopje (Central European Standard Time)",
            "Europe/Sofia (FLE Standard Time)",
            "Europe/Stockholm (W. Europe Standard Time)",
            "Europe/Tallinn (FLE Standard Time)",
            "Europe/Tirane (Central Europe Standard Time)",
            "Europe/Uzhgorod (FLE Standard Time)",
            "Europe/Vaduz (W. Europe Standard Time)",
            "Europe/Vatican (W. Europe Standard Time)",
            "Europe/Vienna (W. Europe Standard Time)",
            "Europe/Vilnius (FLE Standard Time)",
            "Europe/Volgograd (Volgograd Standard Time)",
            "Europe/Warsaw (Central European Standard Time)",
            "Europe/Zagreb (Central European Standard Time)",
            "Europe/Zaporozhye (FLE Standard Time)",
            "Europe/Zurich (W. Europe Standard Time)",
            "Indian/Antananarivo (E. Africa Standard Time)",
            "Indian/Chagos (Central Asia Standard Time)",
            "Indian/Christmas (SE Asia Standard Time)",
            "Indian/Cocos (Myanmar Standard Time)",
            "Indian/Comoro (E. Africa Standard Time)",
            "Indian/Kerguelen (West Asia Standard Time)",
            "Indian/Mahe (Mauritius Standard Time)",
            "Indian/Maldives (West Asia Standard Time)",
            "Indian/Mauritius (Mauritius Standard Time)",
            "Indian/Mayotte (E. Africa Standard Time)",
            "Indian/Reunion (Mauritius Standard Time)",
            "Pacific/Apia (Samoa Standard Time)",
            "Pacific/Auckland (New Zealand Standard Time)",
            "Pacific/Bougainville (Bougainville Standard Time)",
            "Pacific/Chatham (Chatham Islands Standard Time)",
            "Pacific/Easter (Easter Island Standard Time)",
            "Pacific/Efate (Central Pacific Standard Time)",
            "Pacific/Enderbury (UTC+13)",
            "Pacific/Fakaofo (UTC+13)",
            "Pacific/Fiji (Fiji Standard Time)",
            "Pacific/Funafuti (UTC+12)",
            "Pacific/Galapagos (Central America Standard Time)",
            "Pacific/Gambier (UTC-09)",
```

```
            "Pacific/Guadalcanal (Central Pacific Standard Time)",
            "Pacific/Guam (West Pacific Standard Time)",
            "Pacific/Honolulu (Hawaiian Standard Time)",
            "Pacific/Johnston (Hawaiian Standard Time)",
            "Pacific/Kiritimati (Line Islands Standard Time)",
            "Pacific/Kosrae (Central Pacific Standard Time)",
            "Pacific/Kwajalein (UTC+12)",
            "Pacific/Majuro (UTC+12)",
            "Pacific/Marquesas (Marquesas Standard Time)",
            "Pacific/Midway (UTC-11)",
            "Pacific/Nauru (UTC+12)",
            "Pacific/Niue (UTC-11)",
            "Pacific/Norfolk (Norfolk Standard Time)",
            "Pacific/Noumea (Central Pacific Standard Time)",
            "Pacific/Pago_Pago (UTC-11)",
            "Pacific/Palau (Tokyo Standard Time)",
            "Pacific/Pitcairn (UTC-08)",
            "Pacific/Port_Moresby (West Pacific Standard Time)",
            "Pacific/Rarotonga (Hawaiian Standard Time)",
            "Pacific/Saipan (West Pacific Standard Time)",
            "Pacific/Tahiti (Hawaiian Standard Time)",
            "Pacific/Tarawa (UTC+12)",
            "Pacific/Tongatapu (Tonga Standard Time)",
            "Pacific/Wake (UTC+12)",
            "Pacific/Wallis (UTC+12)",
            "US/Alaska (Alaskan Standard Time)",
            "US/Arizona (US Mountain Standard Time)",
            "US/Central (Central Standard Time)",
            "US/Eastern (Eastern Standard Time)",
            "US/Hawaii (Hawaiian Standard Time)",
            "US/Mountain (Mountain Standard Time)",
            "US/Pacific (Pacific Standard Time)",
            "UTC (UTC)"
          ]
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId",
          "Reboot",
          "TimeZone"
        ]
      },
      "additionalProperties": false,
```

```
      "required": [
        "InstanceId",
        "Reboot",
        "TimeZone"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3gf8dolbo8x9p

Classifications:

- Deployment | Advanced stack components | Database Migration Service (DMS) | Create target endpoint

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create DMS target endpoint",
  "description": "Use to create a Database Migration Service (DMS) target endpoint for
 RDS supported MySQL, MariaDB, PostgreSQL, Oracle and Microsoft SQL server engine.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
```

```
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
```

```
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-knghtmmgefafdq89u",
      "type": "string",
      "enum": [
        "stm-knghtmmgefafdq89u"
      ],
      "default": "stm-knghtmmgefafdq89u"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "CertificateArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) for the certificate to use
with the target. This is required if SslMode = verify-ca or verify-full.",
          "pattern": "^$|^arn:aws:dms:[a-z0-9-]+:[0-9]{12}:cert:[A-Z0-9]+$",
          "default": ""
        },
        "DatabaseName": {
          "type": "string",
          "description": "The name of the target database. Must not be blank if
EngineName = oracle, postgres or sqlserver.",
          "default": ""
        },
        "EndpointIdentifier": {
          "type": "string",
          "description": "The identifier to be used for the target endpoint. This is a
label for the endpoint to help you identify it. It must be unique for all endpoints
owned by your AWS account in the current region. It must begin with a letter, must
```

```
contain only ASCII letters, digits and hyphens and must not end with a hyphen or
contain two consecutive hyphens.",
          "pattern": "^$|^(?!.*--)[a-zA-Z][a-zA-Z0-9-]*[a-zA-Z0-9]$",
          "default": ""
        },
        "EngineName": {
          "type": "string",
          "description": "The type of engine this target endpoint is connected to.
Amazon RDS-supported MySQL, MariaDB, PostgreSQL, Oracle and Microsoft SQL are the
options.",
          "enum": [
            "mariadb",
            "mysql",
            "oracle",
            "postgres",
            "sqlserver"
          ]
        },
        "ExtraConnectionAttributes": {
          "type": "string",
          "description": "Additional attributes associated with the connection.
For example, to disable foreign key checks in MySQL compatible database as targets
add initstmt=SET FOREIGN_KEY_CHECKS=0. See 'Targets for Data Migration' in AWS DMS
documentation.",
          "default": ""
        },
        "KmsKeyId": {
          "type": "string",
          "description": "This is the customer master key (CMK) that is used to encrypt
connection parameters. If not specified the default CMK for AWS DMS is used.",
          "default": ""
        },
        "Password": {
          "type": "string",
          "description": "The password to be used to log in to the endpoint
database.",
          "metadata": {
            "ams:sensitive": true
          },
          "default": ""
        },
        "Port": {
          "type": "string",
          "description": "The port used by the endpoint database.",
```

```
          "pattern": "^$|^([1-9]|[1-8][0-9]|9[0-9]|[1-8][0-9]{2}|9[0-8][0-9]|99[0-9]|
[1-8][0-9]{3}|9[0-8][0-9]{2}|99[0-8][0-9]|999[0-9]|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4]
[0-9]{2}|655[0-2][0-9]|6553[0-5])$",
          "default": ""
        },
        "ServerName": {
          "type": "string",
          "description": "The name of the server where the target database resides.
 For an EC2 instance, this can be the IP address or the hostname. For an Amazon RDS DB
 instance, this can be the endpoint for the DB instance.",
          "default": ""
        },
        "SslMode": {
          "type": "string",
          "description": "The SSL mode to ecrypt connections for target endpoint.
 Not all SSL modes work with all database endpoints. See 'Using SSL' in AWS DMS
 documentation.",
          "enum": [
            "none",
            "require",
            "verify-ca",
            "verify-full"
          ],
          "default": "none"
        },
        "Username": {
          "type": "string",
          "description": "The user name to be used to log in to the target database.",
          "metadata": {
            "ams:sensitive": true
          },
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "EndpointIdentifier",
          "EngineName",
          "ServerName",
          "Port",
          "Username",
          "Password",
          "DatabaseName",
          "ExtraConnectionAttributes",
```

```
          "KmsKeyId",
          "SslMode",
          "CertificateArn"
        ]
      },
      "required": [
        "EngineName",
        "ServerName",
        "Port",
        "Username",
        "Password"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "VpcId",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3gg0id58rn82h

Classifications:

- Management | Advanced stack components | EBS snapshot | Share

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Share EBS Snapshot",
  "description": "Share an Elastic Block Store (EBS) snapshot with another AMS account.
 If the destination account is onboarded in a different AMS Region, use change type ID
 ct-3lkbpansfv69k in the destination account to copy shared snapshot across regions.
 Only snapshots encrypted with managed KMS keys can be shared.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-ShareEBSSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-ShareEBSSnapshot"
      ],
      "default": "AWSManagedServices-ShareEBSSnapshot"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AccountId": {
          "description": "The ID of the AWS account the EBS snapshots will be shared
 with, in the form 123456789012.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{12}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SnapshotId": {
          "description": "The ID of the EBS snapshot to share.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$"
          },
```

```
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SnapshotId",
          "AccountId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SnapshotId",
        "AccountId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3gjfayulf5hhs

Classifications:

- Management | Managed account | Developer mode | Enable (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Developer Mode",
```

```
 "description": "Enable Developer Mode (Dev Mode). Dev mode provides you with elevated
permissions, in AMS Plus accounts, to provision and update AWS resources outside
of the AMS change management process. Dev mode does this by leveraging native AWS
API calls within the AMS Virtual Private Cloud (VPC), enabling you to design and
implement infrastructure and applications in your managed environment. When using
an account that has Dev mode enabled, continuity management, patch management, and
change management are provided for resources provisioned through the AMS change
management process or by using an AMS Amazon Machine Image (AMI). However, these
AMS management features are not offered for resources provisioned through native
AWS APIs. Rather, you are responsible for monitoring infrastructure resources that
are provisioned outside of the AMS change management process. Dev mode is limited
to accounts with non-production workloads. With elevated permissions, you have an
increased responsibility to ensure adherence to internal controls.",
 "type": "object",
 "properties": {
   "Enable": {
     "description": "To confirm that you are enabling Dev mode, enter Yes. If this
parameter is left unspecified, Dev mode is not enabled.",
     "type": "string",
     "enum": [
       "Yes"
     ]
   },
   "Priority": {
     "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
     "type": "string",
     "enum": [
       "Low",
       "Medium",
       "High"
     ]
   }
 },
 "additionalProperties": false,
 "metadata": {
   "ui:order": [
     "Enable",
     "Priority"
   ]
 },
 "required": [
   "Enable"
 ]
```

```
}
```

Schema for Change Type ct-3glr80c15rp7z

Classifications:

- Management | Standalone resources | RDS instance | Terminate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Terminate Standalone DB Instance Or Cluster",
  "description": "Terminate a standalone DB instance or cluster. The automation checks
 that the DB instance, or cluster, is not part of a CloudFormation stack and does not
 have termination protection enabled. Please note that deleting the DB cluster deletes
 all the automated backups for that DB cluster and those backups can't be recovered.
 Standalone resources for testing purposes are created by AMS upon your request, they
 are not part of a stack and they can't be deleted with ct-0q0bic0ywqk6c.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
TerminateStandaloneDBInstanceOrCluster.",
      "type": "string",
      "enum": [
        "AWSManagedServices-TerminateStandaloneDBInstanceOrCluster"
      ],
      "default": "AWSManagedServices-TerminateStandaloneDBInstanceOrCluster"
    },
    "Region": {
      "description": "The AWS Region where DB identifier is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Confirmation": {
      "description": "Explicitly confirm the termination of the DB identifier with
 'permanently delete', note that the RFC is not created if this parameter is null.
 Additionally, once the DB identifier is deleted it can't be restored unless there is
 an snapshot for the DB identifier.",
      "type": "string",
      "pattern": "^permanently delete$"
    },
```

```
    "Parameters": {
      "type": "object",
      "properties": {
        "CreateFinalSnapshot": {
          "description": "True to create a final DB snapshot before deleting the DB
 identifier, false to not create a final snapshot. By default, the DB snapshot is
 created. If set to false and there are no existing snapshots for the DB identifier, it
 can't be restored.",
          "type": "boolean",
          "default": true
        },
        "DBIdentifierArn": {
          "description": "The Amazon Resource Name (ARN) that uniquely identifies the
 DB instance or cluster.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{12}:(db|cluster):[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
        },
        "DeleteAutomatedBackups": {
          "description": "True to remove automated (system) backups immediately after
 the DB instance is deleted, false to not remove the backups immediately; applies to DB
 instance backups only. Default is false. Note that automated  backups are deleted once
 the snapshot expires, based on the retention period settings the source instance had
 when you deleted it. Retained automated backups are removed by the system after their
 last system snapshot expires.",
          "type": "boolean",
          "default": false
        },
        "FinalDBSnapshotIdentifier": {
          "description": "A meaningful name for the DB identifier snapshot to be
 created when the CreateFinalSnapshot parameter is set to true.",
          "type": "string",
          "pattern": "^[a-zA-Z](?!.*--)[a-zA-Z0-9-]*[a-zA-Z0-9]$|^$",
          "minLength": 0,
          "maxLength": 256,
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "DBIdentifierArn",
          "DeleteAutomatedBackups",
          "CreateFinalSnapshot",
          "FinalDBSnapshotIdentifier"
```

```
        ]
      },
      "additionalProperties": false,
      "required": [
        "DBIdentifierArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Confirmation",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Confirmation",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3hox8uwjgze1f

Classifications:

- Deployment | Advanced stack components | Identity and Access Management (IAM) | Create SAML identity provider

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create SAML Identity Provider",
  "description": "Create an IAM identity provider using the SAML metadata document file
 that you stored in your chosen S3 bucket.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-HandleCreateSamlProvider-Admin",
```

```
      "type": "string",
      "enum": [
        "AWSManagedServices-HandleCreateSamlProvider-Admin"
      ],
      "default": "AWSManagedServices-HandleCreateSamlProvider-Admin"
    },
    "Region": {
      "description": "The AWS Region of the account, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SAMLMetadataDocumentURL": {
          "description": "The S3 URL of the SAML metadata document file, in the form
s3://bucketname/path/to/saml-metadata.xml.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^s3://[a-z0-9]([-.a-z0-9]+)[a-z0-9]/.+$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Name": {
          "description": "A meaningful name for the identity provider.",
          "type": "array",
          "items": {
            "type": "string",
            "default": "customer-saml",
            "pattern": "^[\\w._-]{1,128}"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SAMLMetadataDocumentURL",
          "Name"
        ]
      },
      "required": [
```

```
        "SAMLMetadataDocumentURL"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3j2zstluz6dxq

Classifications:

- Management | Advanced stack components | Security group | Authorize ingress rule

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Authorize Ingress Rule",
  "description": "Authorize the ingress rule for the specified security group (SG). You
 must specify the configurations of the ingress rule that you are authorizing. Note
 that this adds an ingress rule to the specified SG but does not modify any existing
 ingress rules.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-
AuthorizeSecurityGroupIngressRuleV3.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AuthorizeSecurityGroupIngressRuleV3"
      ],
```

```
      "default": "AWSManagedServices-AuthorizeSecurityGroupIngressRuleV3"
    },
    "Region": {
      "description": "The AWS Region in which the security group is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SecurityGroupId": {
          "description": "The ID of the security group (SG) that you are updating, in
 the form sg-0123456789abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[0-9a-f]{8}$|^sg-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IpProtocol": {
          "description": "The IP protocol name, or IP protocol number, for the ingress
 rule. For example, for TCP, enter either TCP, or (IP protocol number) 6. If you enter
 ICMP, you can specify any or all of the ICMP types and codes.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\+-\\\\(\\\\)\\w]{1,18}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "FromPort": {
          "description": "Start of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
```

```
          "maxItems": 1
        },
        "ToPort": {
          "description": "End of allowed port range, from 0 to 65535 for TCP/UDP. For
 ICMP, use -1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^-1$|^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4]
[0-9]{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Source": {
          "description": "An IP address range in CIDR notation, in the form
 255.255.255.255/32; or the ID of another security group in the same Region; or self,
 to specify the same security group.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8,17}$|^self$|^pl-\\w+|^[0-9]{12}\\/sg-[0-9a-f]{8,17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Description": {
          "description": "A meaningful description of the ingress rule.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^$|^[ a-zA-Z0-9._\\-:/()#,@\\[\\]+=&;{}!$\\*]{1,255}$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "SecurityGroupId",
          "IpProtocol",
          "FromPort",
```

```
          "ToPort",
          "Source",
          "Description"
        ]
      },
      "required": [
        "SecurityGroupId",
        "IpProtocol",
        "FromPort",
        "ToPort",
        "Source"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3jo8yccbin4it

Classifications:

- Management | Managed landing zone | Networking account | Disassociate TGW attachment

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Disassociate TGW Attachment",
  "description": "Disassociate transit gateway (TGW) attachment from the transit
 gateway (TGW) route table. Use this change type for multi-account landing zone (MALZ)
 in Networking account only.",
```

```
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-DisassociateTGWAttachment.",
     "type": "string",
     "enum": [
       "AWSManagedServices-DisassociateTGWAttachment"
     ],
     "default": "AWSManagedServices-DisassociateTGWAttachment"
   },
   "Region": {
     "description": "The AWS Region in which the TGW attachment and TGW route table is
located, in the form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "TransitGatewayAttachmentId": {
         "description": "The ID of the TGW attachment to disassociate from the TGW
route table.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^tgw-attach-[a-z0-9]{17}$"
         },
         "maxItems": 1
       },
       "TransitGatewayRouteTableId": {
         "description": "The ID of the TGW route table.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^tgw-rtb-[a-z0-9]{17}$"
         },
         "maxItems": 1
       }
     },
     "metadata": {
       "ui:order": [
         "TransitGatewayAttachmentId",
         "TransitGatewayRouteTableId"
       ]
```

```
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayAttachmentId",
        "TransitGatewayRouteTableId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3jrqmeq7j0wke

Classifications:

- Deployment | Advanced stack components | Redshift | Create (cluster from snapshot)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Redshift Cluster From Snapshot",
  "description": "Create a Redshift cluster with the same configration as the source
 snapshot.",
  "type": "object",
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
```

```
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
```

```
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-szovkq00000000000",
      "type": "string",
      "enum": [
        "stm-szovkq00000000000"
      ],
      "default": "stm-szovkq00000000000"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ClusterIdentifier": {
          "type": "string",
          "description": "A unique identifier for the cluster. Only ASCII letters,
digits, hyphens. Cannot end with a hyphen or have more than two consecutive
hyphens.",
          "pattern": "^(|(?!.*--.*)(?!.*-$)[a-z][a-z0-9-]{0,62})$"
        },
        "ClusterSnapshot": {
          "type": "string",
          "description": "The name of the snapshot from which to create the new
cluster. Only ASCII letters, digits, and hyphens.",
          "pattern": "^[a-zA-Z0-9-]{1,255}$|^rs:[a-zA-Z0-9-]{1,255}$"
        },
        "NodeType": {
          "description": "The type of Amazon Redshift cluster node. The node type
determines the CPU, RAM, storage capacity, and storage drive type for each node.
You can only modify this if you are using any AWS DS (dense storage) node type. In
that case, you can choose to restore into another DS node type of the same size. For
example, you can restore ds1.8xlarge into ds2.8xlarge, or ds1.xlarge into ds2.xlarge.
If you have a DC instance type, you must restore into that same instance type and
size.",
          "type": "string",
```

```
          "enum": [
            "ds2.xlarge",
            "ds2.8xlarge",
            "dc2.large",
            "dc2.8xlarge",
            "dc1.large",
            "dc1.8xlarge",
            "ra3.xlplus",
            "ra3.4xlarge",
            "ra3.16xlarge"
          ]
        },
        "SnapshotAccountOwner": {
          "type": "string",
          "description": "The AWS customer account used to create or copy the snapshot.
 Required if you are restoring a snapshot you do not own, optional if you own the
 snapshot. Numbers only, no hyphens.",
          "pattern": "^(|[0-9]{12})$"
        },
        "SnapshotClusterIdentifier": {
          "type": "string",
          "description": "The name of the cluster the source snapshot was created
 from. This parameter is required if your IAM user has a policy containing a snapshot
 resource element that specifies anything other than * for the cluster name.",
          "pattern": "^(|(?!.*--.*)(?!.*-$)[a-z][a-z0-9-]{0,62})$"
        },
        "IamRoles": {
          "type": "string",
          "description": "A comma-delimited list of up to 10 AWS Identity and
 Access Management (IAM) roles that the cluster can use to access other AWS
 services. Supply the IAM roles by their Amazon Resource Name (ARN), in the form
 arn:aws:iam::000000000000:role/customer_redshift_role. The role name must be prefixed
 with \"customer\". Leave blank to not attach any roles to the cluster.",
          "pattern": "^(arn:aws:iam::[0-9]{12}:role/customer[/\\w+=,.@-]+)
(,arn:aws:iam::[0-9]{12}:role/customer[/\\w+=,.@-]+){0,9}$|^$"
        },
        "ParameterGroupName": {
          "type": "string",
          "description": "The name of an existing Amazon Redshift parameter group. If
 no value is provided the default parameter group will be used.",
          "pattern": "^(|[a-zA-Z]+(?:-?[a-zA-Z0-9.]+){1,255})$"
        },
        "ClusterSubnetGroup": {
          "type": "string",
```

```
          "description": "The name of an existing Amazon Redshift subnet group.",
          "pattern": "^[a-zA-Z0-9._-]{1,255}$"
        },
        "AllowVersionUpgrade": {
          "type": "string",
          "description": "True to apply upgrades to the engine that is running on the
cluster, during the maintenance window; false to not.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "SecurityGroups": {
          "type": "array",
          "description": "The identifiers of the security groups to control traffic to
and from the Redshift cluster.",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 5,
          "uniqueItems": true
        },
        "DatabasePortNumber": {
          "type": "integer",
          "description": "The port number on which the cluster accepts incoming
connections.",
          "default": 5439,
          "minimum": 1150,
          "maximum": 65535
        },
        "AutomatedSnapshotRetentionPeriod": {
          "type": "integer",
          "description": "The number of days that automated snapshots are retained. The
default is to retain 7 days of snapshots, and the maximum value is 35 days. To disable
automated snapshot retention, use 0.",
          "default": 7,
          "minimum": 0,
          "maximum": 35
        },
        "PreferredMaintenanceWindow": {
          "type": "string",
```

```
          "description": "The weekly time range (in UTC) during which automated cluster
maintenance can occur. The format of the time range is ddd:hh24:mi-ddd:hh24:mi. Leave
blank to allow Amazon Redshift to choose a random 30 minute maintenance window.",
          "pattern": "^[a-z]{3}:[0-9]{2}:[0-9]{2}-[a-z]{3}:[0-9]{2}:[0-9]{2}$|^$",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "ClusterIdentifier",
          "ClusterSnapshot",
          "SnapshotAccountOwner",
          "SnapshotClusterIdentifier",
          "NodeType",
          "IamRoles",
          "ParameterGroupName",
          "ClusterSubnetGroup",
          "AllowVersionUpgrade",
          "SecurityGroups",
          "DatabasePortNumber",
          "AutomatedSnapshotRetentionPeriod",
          "PreferredMaintenanceWindow"
        ]
      },
      "required": [
        "ClusterSnapshot",
        "ClusterSubnetGroup",
        "NodeType"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags",
      "Parameters"
    ]
  },
  "required": [
```

```
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "StackTemplateId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3jx80fquylzhf

Classifications:

- Management | Advanced stack components | RDS database stack | Update enhanced monitoring

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Enhanced Monitoring",
  "description": "Update the Enhanced Monitoring property of an Amazon Relational
 Database Service (RDS) database instance or cluster. Enhanced Monitoring allows you
 to collect vital operating system metrics and process information, at the defined
 granularity.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-UpdateRDSEnhancedMonitoring.",
      "type": "string",
      "enum": [
        "AWSManagedServices-UpdateRDSEnhancedMonitoring"
      ],
      "default": "AWSManagedServices-UpdateRDSEnhancedMonitoring"
    },
    "Region": {
      "description": "The AWS Region in which the AWS resource is located, in the form
 us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
```

```
        "DBIdentifierArn": {
          "description": "The Amazon Resource Name (ARN) of the RDS instance or
 cluster.",
          "type": "string",
          "pattern": "^arn:(aws|aws-cn|aws-us-gov):rds:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{12}:(db|cluster):[a-zA-Z]{1}(?!.*--)(?!.*-$)[A-Za-z0-9-]{0,62}$"
        },
        "MonitoringInterval": {
          "description": "The interval, in seconds, between points when Enhanced
 Monitoring metrics are collected for the DB instance. The valid intervals are 0, 1, 5,
 10, 15, 30 and 60. To disable collecting Enhanced Monitoring metrics, specify 0.",
          "type": "string",
          "enum": [
            "0",
            "1",
            "5",
            "10",
            "15",
            "30",
            "60"
          ]
        },
        "MonitoringRoleName": {
          "description": "The name of the IAM role that permits RDS to send enhanced
 monitoring metrics to Amazon CloudWatch Logs. If no role is specified, the default
 role 'rds-monitoring-role' will be used or created if it does not exist.",
          "type": "string",
          "default": "rds-monitoring-role",
          "pattern": "^[a-zA-Z0-9_+=,.@-]{1,64}$"
        }
      },
      "metadata": {
        "ui:order": [
          "DBIdentifierArn",
          "MonitoringInterval",
          "MonitoringRoleName"
        ]
      },
      "required": [
        "DBIdentifierArn",
        "MonitoringInterval"
      ],
      "additionalProperties": false
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3kh1wiizlne1i

Classifications:

- Management | Access | Stack read-only access | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Grant Stack Read-Only access",
  "description": "Request read only access for one or more users for one or more
 stacks. The maximum access time is 12 hours.",
  "type": "object",
  "properties": {
    "DomainFQDN": {
      "description": "The FQDN for the user accounts to grant access to.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "StackIds": {
      "description": "A minimum of one stack ID is required.",
      "type": "array",
      "items": {
        "type": "string",
        "pattern": "^stack-[a-z0-9]{17}$"
      },
```

```
      "minItems": 1,
      "uniqueItems": true
    },
    "TimeRequestedInHours": {
      "description": "The amount of time, in hours, requested for access to the
 instance. Access is terminated after this time.",
      "type": "integer",
      "minimum": 1,
      "default": 1
    },
    "Usernames": {
      "description": "One or more Active Directory user names used to grant access.",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "uniqueItems": true
    },
    "VpcId": {
      "description": "The ID of the VPC that contains the stacks where access is
 required, in the form of vpc-12345678 or vpc-1234567890abcdef0.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    }
  },
  "metadata": {
    "ui:order": [
      "VpcId",
      "StackIds",
      "Usernames",
      "DomainFQDN",
      "TimeRequestedInHours"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DomainFQDN",
    "StackIds",
    "Usernames",
    "VpcId"
  ]
}
```

Schema for Change Type ct-3kinq0u4l33zf

Classifications:

- Management | Custom Stack | Stack from CloudFormation Template | Remediate drift
- Management | Standard stacks | Stack | Remediate drift


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Remediate Stack Drift",
  "description": "Remediate the drift (out-of-band changes) in a stack, bringing the
 stack in sync and enabling you to perform future updates using the available Update
 CTs. Note: up to 10 drifted resources will be remediated per RFC.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartDriftRemediation.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartDriftRemediation"
      ],
      "default": "AWSManagedServices-StartDriftRemediation"
    },
    "Region": {
      "description": "The AWS Region in which the CloudFormation stack is located, in
 the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "StackName": {
          "description": "The name of the stack to remediate the drift for, in the form
 of stack-a1b2c3d4e5f67890e.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^stack-[a-z0-9]{8}$|^stack-[a-z0-9]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
```

```
      },
      "DryRun": {
        "description": "True to perform drift remediation in dry run mode, false to
perform drift remediation not in dry run mode. Default is false. Dry run mode checks
if the stack drift can be remediated or not, but does not attempt remediation. Note
that, when DryRun=true, reserved stack outputs for drift remediation, in the form of
AMSCFNDriftRemediationBuildReferences95556500d5, can be added or updated. To learn
more about outputs, see AWS CloudFormation documentation.",
        "type": "array",
        "items": {
          "type": "string",
          "default": "false",
          "enum": [
            "true",
            "false"
          ]
        },
        "minItems": 1,
        "maxItems": 1
      }
    },
    "metadata": {
      "ui:order": [
        "StackName",
        "DryRun"
      ]
    },
    "additionalProperties": false,
    "required": [
      "StackName"
    ]
  }
},
"metadata": {
  "ui:order": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
},
"additionalProperties": false,
"required": [
  "DocumentName",
  "Region",
```

```
    "Parameters"
  ]
}
```

Schema for Change Type ct-3l14e139i5p50

Classifications:

- Deployment | Advanced stack components | ACM Certificate with additional SANs | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "acm-certificate-with-additional-sans",
  "description": "ACM Certificate with additional SANs",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Stack's purpose description",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the vpc to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to 40 tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
```

```
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "pattern": "^[a-zA-Z0-9\\s_./=+-]{1,127}$",
            "minLength": 1,
            "maxLength": 127
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 40,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-[a-z]{17}",
      "type": "string",
      "enum": [
        "stm-ftu71ma6q29bvulv0"
      ]
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60
    },
    "Parameters": {
```

```
    "type": "object",
    "properties": {
      "DomainName": {
        "type": "string",
        "description": "Fully qualified domain name (FQDN), such as www.example.com,
of the site that you want to secure with the ACM certificate. A wildcard can be used
to create a certificate for multiple subdomains, e.g. *.example.com",
        "pattern": "^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$"
      },
      "ValidationDomain": {
        "type": "string",
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the DomainName value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeName1": {
        "type": "string",
        "description": "FQDNs to be included in the Subject Alternative Name
extension of the ACM certificate.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeName2": {
        "type": "string",
        "description": "FQDNs to be included in the Subject Alternative Name
extension of the ACM certificate.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeName3": {
        "type": "string",
        "description": "FQDNs to be included in the Subject Alternative Name
extension of the ACM certificate.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeName4": {
        "type": "string",
        "description": "FQDNs to be included in the Subject Alternative Name
extension of the ACM certificate.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
```

```
      },
      "SubjectAlternativeName5": {
        "type": "string",
        "description": "FQDNs to be included in the Subject Alternative Name
extension of the ACM certificate.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeNameValidationDomain1": {
        "type": "string",
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the SubjectAlternativeName1 value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeNameValidationDomain2": {
        "type": "string",
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the SubjectAlternativeName2 value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeNameValidationDomain3": {
        "type": "string",
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the SubjectAlternativeName3 value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeNameValidationDomain4": {
        "type": "string",
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the SubjectAlternativeName4 value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      },
      "SubjectAlternativeNameValidationDomain5": {
        "type": "string",
```

```
        "description": "The domain that domain name registrars use to send validation
emails. This value must be the same as the domain name or a superdomain of the domain
name. If left blank, the SubjectAlternativeName5 value will be used.",
        "pattern": "^$|^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$",
        "default": ""
      }
    },
    "required": [
      "DomainName"
    ],
    "metadata": {
      "ui:order": [
        "DomainName",
        "ValidationDomain",
        "SubjectAlternativeName1",
        "SubjectAlternativeNameValidationDomain1",
        "SubjectAlternativeName2",
        "SubjectAlternativeNameValidationDomain2",
        "SubjectAlternativeName3",
        "SubjectAlternativeNameValidationDomain3",
        "SubjectAlternativeName4",
        "SubjectAlternativeNameValidationDomain4",
        "SubjectAlternativeName5",
        "SubjectAlternativeNameValidationDomain5"
      ]
    },
    "additionalProperties": false
  }
},
"metadata": {
  "ui:order": [
    "Name",
    "Description",
    "VpcId",
    "StackTemplateId",
    "Parameters",
    "TimeoutInMinutes",
    "Tags"
  ]
},
"required": [
  "Description",
  "VpcId",
  "Name",
```

```
    "StackTemplateId",
    "TimeoutInMinutes",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3lkbpansfv69k

Classifications:

- Deployment | Advanced stack components | EBS snapshot | Copy

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Copy EBS Snapshot",
  "description": "Copy an Elastic Block Store (EBS) snapshot in your AMS account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CopyEBSSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CopyEBSSnapshot"
      ],
      "default": "AWSManagedServices-CopyEBSSnapshot"
    },
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "SourceRegion": {
          "description": "The AWS Region that contains the source snapshot, in the form
 us-east-1.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}$"
```

```
          },
          "minItems": 1,
          "maxItems": 1
        },
        "SourceSnapshotId": {
          "description": "The ID of the EBS snapshot to copy, in the form snap-12345678
 or snap-123456789012345ab.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^snap-[0-9a-f]{8}$|^snap-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "KmsKeyId": {
          "description": "An AWS Key Management Service (KMS) key to encrypt the EBS
 snapshot with. The KMS key is the KMS Key ARN or the KMS key identifier. If left blank
 and the source snapshot is encrypted, the target snapshot will be encrypted using the
 default EBS KMS key.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(arn:aws:kms:[a-z0-9-]+:[0-9]{12}:key/){0,1}[a-f0-9]{8}-[a-
f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}$|^$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Description": {
          "description": "A description for the new snapshot. If left blank a default
 description is used, in the form [Copied {SourceSnapshotId} from {SourceRegion}].",
          "type": "array",
          "items": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
```

```
          "SourceRegion",
          "SourceSnapshotId",
          "KmsKeyId",
          "Description"
        ]
      },
      "additionalProperties": false,
      "required": [
        "SourceRegion",
        "SourceSnapshotId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3ll9hnadql9s1

Classifications:

- Deployment | Advanced stack components | ACM | Create public certificate

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Public ACM Certificate",
  "description": "Create a public AWS Certificate Manager (ACM) certificate with email
 or DNS validation. To create a private ACM certificate, use ct-0hu3q3957aghj.",
  "type": "object",
  "properties": {
    "DocumentName": {
```

```
      "description": "Must be AWSManagedServices-RequestACMCertificateV2",
      "type": "string",
      "enum": [
        "AWSManagedServices-RequestACMCertificateV2"
      ],
      "default": "AWSManagedServices-RequestACMCertificateV2"
    },
    "Region": {
      "description": "The AWS Region in which you want the ACM certificate, in the form
us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "DomainName": {
          "description": "The fully qualified domain name (FQDN), such as
www.example.com, that you want to secure with an ACM certificate.",
          "type": "string",
          "pattern": "^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$"
        },
        "ValidationMethod": {
          "description": "How you will validate that you own or control the domain for
the ACM certificate.",
          "type": "string",
          "enum": [
            "EMAIL",
            "DNS"
          ],
          "default": "EMAIL"
        },
        "CertificateType": {
          "description": "Confirm that you are creating a public ACM certificate. To
create a private ACM certificate, use ct-0hu3q3957aghj.",
          "type": "string",
          "enum": [
            "Public"
          ],
          "default": "Public"
        },
        "ValidationDomain": {
```

```
        "description": "The domain for ACM to use when sending validation emails.
This value must be the same as the DomainName, or a superdomain of the DomainName. If
left blank, the DomainName value is used.",
        "type": "string",
        "pattern": "^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$|^$",
        "default": ""
      },
      "SubjectAlternativeNames": {
        "description": "The additional FQDNs to be included in the subject
alternative name extension of the ACM certificate.",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "^(\\*\\.){0,1}(\\w+)(.\\w+)*(\\.\\w+)$"
        },
        "minItems": 1,
        "maxItems": 5
      },
      "Route53DNSValidation": {
        "description": "True for automatic ACM validation using your Route53 DNS, if
the ACM and the domain are on the same account; false for no automatic validation.
Default is false.",
        "type": "string",
        "enum": [
          "True",
          "False"
        ],
        "default": "False"
      }
    },
    "metadata": {
      "ui:order": [
        "DomainName",
        "CertificateType",
        "ValidationMethod",
        "ValidationDomain",
        "SubjectAlternativeNames",
        "Route53DNSValidation"
      ]
    },
    "additionalProperties": false,
    "required": [
      "DomainName",
      "ValidationMethod"
```

```
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3memthlcmvc1b

Classifications:

- Management | Advanced stack components | Security group | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update a security group",
  "description": "Update the inbound and the outbound rules of a security group, and
 optionally associate it with AWS resources.",
  "type": "object",
  "properties": {
    "SecurityGroupId": {
      "description": "ID of the security group to be updated or disassociated from the
 specified AWS resources.",
      "type": "string",
      "pattern": "^sg-[0-9a-zA-Z]{8}$|^sg-[0-9a-zA-Z]{17}$"
    },
    "AddAssociatedResources": {
      "description": "Additional AWS resources to associate the security group to. For
 example, EC2 instance IDs, RDS DB instance IDs, Load Balancer names, DSM replication
 instance names, EFS mount target IDs, ElastiCache cluster IDs. To remove resources,
 use the Delete Security group CT.",
```

```
      "type": "array",
      "items": {
        "type": "string",
        "minLength": 1,
        "maxLength": 64
      },
      "minItems": 0,
      "maxItems": 10,
      "uniqueItems": true
    },
    "AddInboundRules": {
      "description": "New inbound rules to be added.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
            "description": "A port number or a port range. For example, 80 or
 49152-65535. Use -1 for all ports.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
          },
          "Source": {
            "description": "An IP address, or an IP address range in CIDR notation (for
 example, 203.0.113.5/32), or the ID of another security group in the same region. To
 reference this security group, use self. From behind a firewall, use the public IP
 address or range used by the client computers.",
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8,17}$|^self$|^pl-\\w+|^[0-9]{12}\\/sg-[0-9a-f]{8,17}$"
          },
          "Description": {
            "description": "Meaningful description of the inbound rule.",
```

```
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Source",
            "Description"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Source"
        ]
      },
      "minItems": 0,
      "maxItems": 50
    },
    "RemoveInboundRules": {
      "description": "Existing inbound rules to be removed.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
            "description": "A port number or a port range. For example, 80 or
 49152-65535. Use -1 for all ports.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
```

```
          },
          "Source": {
            "description": "An IP address, or an IP address range in CIDR notation (for
 example, 203.0.113.5/32), or the ID of another security group in the same region. To
 reference this security group, use self. From behind a firewall, use the public IP
 address or range used by the client computers.",
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){0,1}$|^sg-
[0-9a-f]{8,17}$|^self$|^pl-\\w+|^[0-9]{12}\\/sg-[0-9a-f]{8,17}$"
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Source"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Source"
        ]
      },
      "minItems": 0,
      "maxItems": 50
    },
    "AddOutboundRules": {
      "description": "New outbound rules to be added.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
```

```
            "description": "A port number or a port range. For example, 80 or
 49152-65535. Use -1 for all ports.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
          },
          "Destination": {
            "description": "An IP address, or an IP address range in CIDR notation (for
 example, 203.0.113.5/32), or the ID of another security group in the same region. To
 reference this security group, use self. From behind a firewall, use the public IP
 address or range used by the client computers.",
            "type": "string",
            "minLength": 1,
            "maxLength": 64
          },
          "Description": {
            "description": "Meaningful description of the outbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Destination",
            "Description"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Destination"
        ]
      },
      "minItems": 0,
      "maxItems": 50
    },
    "RemoveOutboundRules": {
      "description": "Existing outbound rules to be removed.",
      "type": "array",
```

```
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol name or protocol number for the rule. For
 example, for TCP, it could be protocol name TCP or protocol number 6. If you specify
 ICMP as the protocol, you can specify any or all of the ICMP types and codes.",
            "type": "string",
            "minLength": 1,
            "maxLength": 32
          },
          "PortRange": {
            "description": "A port number or a port range. For example, 80 or
 49152-65535. Use -1 for all ports.",
            "type": "string",
            "pattern": "^-1$|^[Aa][Ll]{2}$|^(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|6[0-4]
[0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])(-(0|[1-5][0-9]{0,4}|[6-9][0-9]{0,3}|
6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2][0-9]|6553[0-5])){0,1}$"
          },
          "Destination": {
            "description": "An IP address, or an IP address range in CIDR notation (for
 example, 203.0.113.5/32), or the ID of another security group in the same region. To
 reference this security group, use self. From behind a firewall, use the public IP
 address or range used by the client computers.",
            "type": "string",
            "minLength": 1,
            "maxLength": 64
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "PortRange",
            "Destination"
          ]
        },
        "required": [
          "Protocol",
          "PortRange",
          "Destination"
        ]
      },
      "minItems": 0,
```

```
      "maxItems": 50
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    },
    "Tags": {
      "description": "Up to 50 tags (key/value pairs) to categorize the resource.
Overwrites the original tags.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
```

```
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "SecurityGroupId",
      "AddAssociatedResources",
      "AddInboundRules",
      "RemoveInboundRules",
      "AddOutboundRules",
      "RemoveOutboundRules",
      "Priority",
      "Tags"
    ]
  },
  "required": [
    "SecurityGroupId"
  ]
}
```

Schema for Change Type ct-3mlsibqhugrf1

Classifications:

- Deployment | Advanced stack components | EBS snapshot | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create EBS Snapshot",
  "description": "Create an Elastic Block Store (EBS) snapshot from an EBS volume. The
 volume must be attached to an EC2 instance.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateEBSSnapshot.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateEBSSnapshot"
      ],
      "default": "AWSManagedServices-CreateEBSSnapshot"
    },
```

```
    "Region": {
      "description": "The AWS Region to use, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VolumeId": {
          "description": "The ID of the source EBS volume, in the form vol-12345678 or
 vol-123456789012345ab.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^vol-[0-9a-f]{8}$|^vol-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Description": {
          "description": "A description for the new snapshot.",
          "type": "array",
          "items": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "VolumeId",
          "Description"
        ]
      },
      "additionalProperties": false,
      "required": [
        "VolumeId"
      ]
    }
  },
  "metadata": {
```

```
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3mvvt2zkyveqj

Classifications:

- Management | Advanced stack components | EC2 instance stack | Stop

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Stop EC2 Instances",
  "description": "Stop up to 50 running EC2 instances. If you specify an EC2 instance
 that is part of an Auto Scaling group (ASG), the instance is terminated and replaced
 by the ASG. If not part of an ASG, the instance remains stopped, in the account, until
 started or deleted.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StopInstances.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StopInstances"
      ],
      "default": "AWSManagedServices-StopInstances"
    },
    "Region": {
      "description": "The AWS Region where the instances are, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
```

```
   "Parameters": {
     "type": "object",
     "properties": {
       "InstanceIds": {
         "description": "A list of up to 50 EC2 instance IDs, in the form
i-1234567890abcdef0 or i-b188560f.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^i-[a-f0-9]{8}$|^i-[a-f0-9]{17}$"
         },
         "minItems": 1,
         "maxItems": 50,
         "uniqueItems": true
       },
       "ForceStop": {
         "description": "True to stop the instances even if the KMS key used in
encrypting any of the volumes of the instance is non-existent or pending deletion.
False to not stop them if the KMS key is non-existent or pending deletion. Stopping
these sorts of instances is not recommended unless the data on them is not required or
is already backed up, because once stopped, they cannot be started.",
         "type": "array",
         "items": {
           "type": "string",
           "default": "false",
           "enum": [
             "true",
             "false"
           ]
         },
         "minItems": 1,
         "maxItems": 1
       },
       "StopASGInServiceInstances": {
         "description": "True to stop and terminate any ASG instance that is in the
'InService' state. False to only stop standalone instances and ASG instances that are
in the 'Standby' state (ASG instances in the 'InService' state are not stopped. )",
         "type": "array",
         "items": {
           "type": "string",
           "default": "false",
           "enum": [
             "true",
             "false"
```

```
            ]
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "*"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceIds"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3nba0wtdugnan

Classifications:

- Deployment | Directory Service | DNS | Create conditional forwarder

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create AD DNS Conditional Forwarder",
```

```
  "description": "Create AD DNS conditional forwarder with up to five DNS servers
 associated with a remote domain name. For multi-account landing zone (MALZ), use this
 change type in the shared services account.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateADDNSConditionalForwarder-
Admin.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateADDNSConditionalForwarder-Admin"
      ],
      "default": "AWSManagedServices-CreateADDNSConditionalForwarder-Admin"
    },
    "Region": {
      "description": "The AWS Region where the Microsoft AD in Directory Service is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "RemoteDomainName": {
          "description": "The fully qualified domain name (FQDN) of the remote
 domain.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^([a-zA-Z0-9]+[\\.-])+([a-zA-Z0-9])+[.]?$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "IPAddresses": {
          "description": "A list of private IP addresses of the remote DNS servers
 associated with the conditional forwarder.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(10\\.(\\d{1,3})\\.(\\d{1,3})\\.(\\d{1,3}))$|^(192\\.168\\.(\
\d{1,3})\\.(\\d{1,3}))$|^(172\\.(1[6-9]|2[0-9]|3[0-1])\\.[0-9]{1,3}\\.[0-9]{1,3})$"
          },
          "minItems": 1,
```

```
          "maxItems": 5,
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "RemoteDomainName",
          "IPAddresses"
        ]
      },
      "additionalProperties": false,
      "required": [
        "RemoteDomainName",
        "IPAddresses"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3nmhh0qr338q6

Classifications:

- Management | Managed landing zone | Networking account | Associate TGW attachment

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Associate TGW Attachment",
```

```
 "description": "Associate transit gateway (TGW) attachment to the transit gateway
(TGW) route table. Use this change type for multi-account landing zone (MALZ) in
Networking account only.",
 "type": "object",
 "properties": {
   "DocumentName": {
     "description": "Must be AWSManagedServices-AssociateTGWAttachment.",
     "type": "string",
     "enum": [
       "AWSManagedServices-AssociateTGWAttachment"
     ],
     "default": "AWSManagedServices-AssociateTGWAttachment"
   },
   "Region": {
     "description": "The AWS Region in which the TGW attachment and TGW route table is
located, in the form us-east-1.",
     "type": "string",
     "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
   },
   "Parameters": {
     "type": "object",
     "properties": {
       "TransitGatewayAttachmentId": {
         "description": "The ID of the TGW attachment to associate to the TGW route
table.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^tgw-attach-[a-z0-9]{17}$"
         },
         "maxItems": 1
       },
       "TransitGatewayRouteTableId": {
         "description": "The ID of the TGW route table.",
         "type": "array",
         "items": {
           "type": "string",
           "pattern": "^tgw-rtb-[a-z0-9]{17}$"
         },
         "maxItems": 1
       }
     },
     "metadata": {
       "ui:order": [
```

```
          "TransitGatewayAttachmentId",
          "TransitGatewayRouteTableId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayAttachmentId",
        "TransitGatewayRouteTableId"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3oafsdbzjtuqp

Classifications:

- Deployment | Advanced stack components | VPC endpoint (interface) | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create VPC Endpoint (Interface)",
  "description": "Create an interface VPC endpoint, which allows you to connect to
 services powered by AWS PrivateLink, including many AWS services.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
```

```
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component. This becomes the Stack
Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
```

```
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-f0cumpt1rfc1p1739",
      "type": "string",
      "enum": [
        "stm-f0cumpt1rfc1p1739"
      ],
      "default": "stm-f0cumpt1rfc1p1739"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 60,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "VpcId": {
          "type": "string",
          "description": "The VPC ID to attach the interface endpoint to, in the form
vpc-0123abcd or vpc-01234567890abcdef.",
          "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
        },
        "ServiceName": {
          "type": "string",
          "description": "The service name the interface VPC endpoint is for. For
example, com.amazonaws.ap-southeast-2.cloudformation.",
          "pattern": "(com.amazonaws|aws.sagemaker).[a-z0-9-.]{3,60}"
        },
        "SecurityGroups": {
          "type": "array",
          "description": "The security groups to associate with the interface VPC
endpoint, in the form sg-0123abcd or sg-01234567890abcdef.",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
```

```
          },
          "uniqueItems": true
        },
        "SubnetIds": {
          "type": "array",
          "description": "The subnet IDs to associate with the interface VPC endpoint,
in the form subnet-0123abcd or subnet-01234567890abcdef.",
          "items": {
            "type": "string",
            "pattern": "^subnet-[a-z0-9]{8}$|^subnet-[a-z0-9]{17}$"
          },
          "uniqueItems": true
        },
        "EnablePrivateDns": {
          "type": "string",
          "description": "True to associate a private hosted zone with the VPC, false
to not. The private hosted zone contains a record set for the default public DNS name
for the service for the Region, which resolves to the private IP addresses of the
network interfaces that are attached to the interface VPC endpoint.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        }
      },
      "metadata": {
        "ui:order": [
          "VpcId",
          "ServiceName",
          "SecurityGroups",
          "SubnetIds",
          "EnablePrivateDns"
        ]
      },
      "required": [
        "VpcId",
        "ServiceName",
        "SecurityGroups",
        "SubnetIds"
      ],
      "additionalProperties": false
    }
  },
```

```
  "metadata": {
    "ui:order": [
      "VpcId",
      "Name",
      "Description",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags",
      "Parameters"
    ]
  },
  "required": [
    "VpcId",
    "Name",
    "Description",
    "TimeoutInMinutes",
    "StackTemplateId",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3ovo7px2vsa6n

Classifications:

- Management | Advanced stack components | KMS key | Update (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update KMS Key",
  "description": "Request an update of a KMS Key.",
  "type": "object",
  "properties": {
    "KeyDescription": {
      "description": "A meaningful description of the KMS key; for example, a
 description that indicates that the KMS key is appropriate for a task. The default
 value is an empty string (no description). Note that the description appears in the
 details for the key in the KMS console. Do not include confidential or sensitive
 information as this field may appear in plain text in CloudTrail logs and other
 output.",
      "type": "string",
```

```
      "maxLength": 5000
    },
    "TargetKeyARN": {
      "description": "The Amazon Resource Name (ARN) of the target KMS key,in the
 form arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab, to
 update.",
      "type": "string",
      "pattern": "^(arn:(aws|aws-cn|aws-us-gov):kms:[a-z0-9-]+:[0-9]{12}:key/)?([a-
f0-9]{8}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{4}-[a-f0-9]{12}|mrk-[a-z0-9]{32})$"
    },
    "AliasName": {
      "description": "An alias name for the KMS key. The alias name must be unique in
 the AWS account and region, can be up to 256 characters in length, and is limited to
 use characters a-z, A-Z, 0-9, and /_-",
      "type": "string",
      "pattern": "^[a-zA-Z0-9/_-]{1,256}$"
    },
    "KeyStatus": {
      "description": "The KMS key status. Default is Enabled.",
      "type": "string",
      "default": "Enabled",
      "enum": [
        "Enabled",
        "Disabled",
        "Cancel Key Deletion and Enabled",
        "Cancel Key Deletion and Disabled"
      ]
    },
    "KeyRotation": {
      "description": "True if the KMS key should be rotated, false if it should not.",
      "type": "boolean"
    },
    "KeyPermissions": {
      "description": "Detailed information about the key permissions, or a JSON
 policy document to be attached to the key (paste the policy document into the value
 field).",
      "type": "string",
      "maxLength": 5000
    },
    "PolicyAction": {
      "description": "Whether the given 'KeyPermissions' needs to be appended to the
 existing key policy or to replace the key policy entirely. If you want to add a new
 statement block to the existing policy, choose 'Append'. If you want to replace the
 entire policy or update the policy in specific sections, provide the entire policy
```

```
containing desired changes in 'KeyPermissions' and choose 'Replace'. Leave this
parameter blank if 'KeyPermissions' is not to be modified.",
     "type": "string",
     "enum": [
       "Append",
       "Replace"
     ]
   },
   "Tags": {
     "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
     "type": "array",
     "items": {
       "type": "object",
       "properties": {
         "Key": {
           "type": "string",
           "minLength": 1,
           "maxLength": 127
         },
         "Value": {
           "type": "string",
           "minLength": 1,
           "maxLength": 255
         }
       },
       "additionalProperties": false,
       "metadata": {
         "ui:order": [
           "Key",
           "Value"
         ]
       },
       "required": [
         "Key",
         "Value"
       ]
     },
     "minItems": 0,
     "maxItems": 50,
     "uniqueItems": true
   },
   "Operation": {
     "description": "Must be Update.",
     "type": "string",
```

```
      "default": "Update",
      "enum": [
        "Update"
      ]
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "KeyDescription",
      "TargetKeyARN",
      "AliasName",
      "KeyStatus",
      "KeyRotation",
      "KeyPermissions",
      "PolicyAction",
      "Tags",
      "Operation",
      "Priority"
    ]
  },
  "required": [
    "TargetKeyARN",
    "Operation"
  ]
}
```

Schema for Change Type ct-3oy53m1qzl2s5

Classifications:

- Management | Patching | On demand patching | Run

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "On Demand Patching",
  "description": "Run on-demand SSM patching on specified instances; either a list of
 instances or instances with the specified tag/key pair.",
  "additionalProperties": false,
  "properties": {
    "Description": {
      "description": "A meaningful description for this on demand patch run.",
      "maxLength": 500,
      "minLength": 1,
      "type": "string"
    },
    "Name": {
      "description": "A friendly name for this on demand patch run.",
      "maxLength": 128,
      "minLength": 3,
      "type": "string"
    },
    "StartInactiveInstances": {
      "description": "True to start instances that were stopped before being patched,
 false to keep them stopped. Allowed values are \"True\" and \"False\".",
      "enum": [
        "True",
        "False"
      ],
      "type": "string"
    },
    "BackupVaultName": {
      "description": "The name of a logical container where backups are stored.
 The backup vault name is case sensitive and must contain from 2 to 50 alphanumeric
 characters or hyphens.",
      "default": "ams-manual-backups",
      "pattern": "^[a-zA-Z0-9\\_\\-]{2,50}$",
      "type": "string"
    },
    "BackupIamRole": {
      "description": "The name of the role that allows AWS Backup to perform
 the actions on your behalf. The backup IAM role name must contain from 1 to 64
 alphanumeric characters or hyphens.",
      "default": "ams-backup-iam-role",
      "pattern": "^[a-zA-Z0-9\\_\\-]{1,64}$",
      "type": "string"
```

```
    },
    "BackupRetentionInDays": {
      "description": "The number of days the backup taken before patching will remain
available.",
      "default": "21",
      "pattern": "^([1-9]|[1-9][0-9]|[1-2][0-9]{2}|3[0-5][0-9]{1}|36[0-4]|365)$",
      "type": "string"
    },
    "PatchingTargets": {
      "description": "EC2 instances to run on-demand patching.",
      "items": {
        "additionalProperties": false,
        "properties": {
          "Key": {
            "description": "Enter \"InstanceIds\" to patch instances based on
instanceIds. Or \"tag:\" followed by the tag key, such as \"tag:Patch Group\". The
instances with whatever key/value pair that you enter, are marked for on-demand
patching.",
            "maxLength": 150,
            "minLength": 1,
            "type": "string",
            "pattern": "^(tag:[\\w\\s_.:/=+\\-@]+|InstanceIds)$"
          },
          "Values": {
            "description": "Provide the list of instanceIds if the key mentioned
above is \"InstanceIds\". Else, provide a single tag value corresponding to the
\"Key\" mentioned above. See AWS Systems Manager, Automation queue, documentation for
information on queue limits.",
            "items": {
              "maxLength": 255,
              "minLength": 1,
              "type": "string"
            },
            "minItems": 1,
            "type": "array",
            "uniqueItems": true
          }
        },
        "metadata": {
          "ui:order": [
            "Key",
            "Values"
          ]
        },
```

```
        "required": [
          "Key",
          "Values"
        ],
        "type": "object"
      },
      "maxItems": 1,
      "minItems": 1,
      "type": "array"
    }
  },
  "metadata": {
    "ui:order": [
      "Name",
      "Description",
      "PatchingTargets",
      "StartInactiveInstances",
      "BackupVaultName",
      "BackupIamRole",
      "BackupRetentionInDays"
    ]
  },
  "required": [
    "Name",
    "PatchingTargets",
    "StartInactiveInstances"
  ],
  "type": "object"
}
```

Schema for Change Type ct-3pc215bnwb6p7

Classifications:

- Deployment | Advanced stack components | Security group | Create

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Security Group",
  "description": "Create a security group with limited scope. For complex security
 groups, use the manual Security group Create change type (ct-1oxx2g2d7hc90).",
  "type": "object",
```

```
  "properties": {
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "SecurityGroupName": {
      "description": "A name for the security group. The name cannot start with \"sg-
\", and must be unique within the VPC.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "SecurityGroupDescription": {
      "description": "Meaningful information about the security group.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "TcpUdpIngressRules": {
      "description": "TCP and UDP based ingress rules for the security group. No
 inbound TCP or UDP traffic originating from another host to your instance is allowed
 until you add these rules to the security group.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol for this rule, either TCP or UDP. Note you can
 add multiple rules for each.",
            "type": "string",
            "enum": [
              "TCP",
              "UDP"
            ]
          },
          "FromPort": {
            "description": "Start of allowed port range (0-65535 for TCP/UDP).",
            "type": "integer",
            "minimum": 0,
            "maximum": 65535
          },
          "ToPort": {
```

```
            "description": "End of allowed port range (0-65535 for TCP/UDP).",
            "type": "integer",
            "minimum": 0,
            "maximum": 65535
          },
          "Description": {
            "description": "Meaningful description of the TCP/UDP inbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          },
          "AddressRanges": {
            "description": "An IP address range in CIDR notation (for example,
 10.0.0.0/8). If you want to specify a single IP, use a CIDR Prefix of \"/32\". You
 must specify either AddressRanges parameter or SecurityGroupIds parameter, but you can
 also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^self$|^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.)
{3}([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){1}$",
              "minLength": 1,
              "maxLength": 64
            }
          },
          "SecurityGroupIds": {
            "description": "The ID of another security group in the same Region. To use
 this security group, specify \"self\". You must specify either AddressRanges parameter
 or SecurityGroupIds parameter, but you can also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$|^self$",
              "minLength": 1,
              "maxLength": 64
            }
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "FromPort",
            "ToPort",
```

```
            "AddressRanges",
            "SecurityGroupIds",
            "Description"
          ]
        },
        "required": [
          "Protocol",
          "FromPort",
          "ToPort"
        ]
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "TcpUdpEgressRules": {
      "description": "TCP and UDP based outbound rules for the security group. Unless
custom egress rules are specified, all TCP and UDP outbound traffic originating from
your instance is allowed.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Protocol": {
            "description": "The protocol for this rule, either TCP or UDP. Note you can
add multiple rules for each.",
            "type": "string",
            "enum": [
              "TCP",
              "UDP"
            ]
          },
          "FromPort": {
            "description": "Start of allowed port range (0-65535 for TCP/UDP).",
            "type": "integer",
            "minimum": 0,
            "maximum": 65535
          },
          "ToPort": {
            "description": "End of allowed port range (0-65535 for TCP/UDP).",
            "type": "integer",
            "minimum": 0,
            "maximum": 65535
          },
```

```
          "Description": {
            "description": "Meaningful description of the TCP/UDP outbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          },
          "AddressRanges": {
            "description": "An IP address range in CIDR notation (for example,
 10.0.0.0/8). If you want to specify a single IP, use a CIDR Prefix of \"/32\". You
 must specify either AddressRanges parameter or SecurityGroupIds parameter, but you can
 also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^self$|^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.)
{3}([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){1}$",
              "minLength": 1,
              "maxLength": 64
            }
          },
          "SecurityGroupIds": {
            "description": "The ID of another security group in the same Region. To use
 this security group, specify \"self\". You must specify either AddressRanges parameter
 or SecurityGroupIds parameter, but you can also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$|^self$",
              "minLength": 1,
              "maxLength": 64
            }
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Protocol",
            "FromPort",
            "ToPort",
            "AddressRanges",
            "SecurityGroupIds",
            "Description"
          ]
        },
```

```
        "required": [
          "Protocol",
          "FromPort",
          "ToPort"
        ]
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "IcmpIngressRules": {
      "description": "ICMP based ingress rules for the security group. No inbound ICMP
 traffic originating from another host to your instance is allowed until you add these
 rules to the security group.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Type": {
            "description": "The ICMP type. Specify \"-1\" for all types.",
            "type": "integer"
          },
          "Code": {
            "description": "The ICMP code. Specify \"-1\" for all codes. Must be \"-1\"
 if ICMP type is \"-1\".",
            "type": "integer"
          },
          "Description": {
            "description": "Meaningful description of the ICMP inbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          },
          "AddressRanges": {
            "description": "An IP address range in CIDR notation (for example,
 10.0.0.0/8). If you want to specify a single IP, use a CIDR Prefix of \"/32\". You
 must specify either AddressRanges parameter or SecurityGroupIds parameter, but you can
 also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^self$|^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.)
{3}([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){1}$",
              "minLength": 1,
```

```
            "maxLength": 64
          }
        },
        "SecurityGroupIds": {
          "description": "The ID of another security group in the same Region. To use
this security group, specify \"self\". You must specify either AddressRanges parameter
or SecurityGroupIds parameter, but you can also specify both.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$|^self$",
            "minLength": 1,
            "maxLength": 64
          }
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "Type",
          "Code",
          "AddressRanges",
          "SecurityGroupIds",
          "Description"
        ]
      },
      "required": [
        "Type",
        "Code"
      ]
    },
    "minItems": 0,
    "maxItems": 60,
    "uniqueItems": true
  },
  "IcmpEgressRules": {
    "description": "ICMP based outbound rules for the security group. Unless custom
egress rules are specified, all ICMP outbound traffic originating from your instance
is allowed.",
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "Type": {
```

```
            "description": "The ICMP type. Specify \"-1\" for all types.",
            "type": "integer"
          },
          "Code": {
            "description": "The ICMP code. Specify \"-1\" for all codes. Must be \"-1\"
 if ICMP type is \"-1\".",
            "type": "integer"
          },
          "Description": {
            "description": "Meaningful description of the ICMP outbound rule.",
            "type": "string",
            "minLength": 0,
            "maxLength": 255
          },
          "AddressRanges": {
            "description": "An IP address range in CIDR notation (for example,
 10.0.0.0/8). If you want to specify a single IP, use a CIDR Prefix of \"/32\". You
 must specify either AddressRanges parameter or SecurityGroupIds parameter, but you can
 also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^self$|^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])\\.)
{3}([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(\\/([0-9]|[1-2][0-9]|3[0-2])){1}$",
              "minLength": 1,
              "maxLength": 64
            }
          },
          "SecurityGroupIds": {
            "description": "The ID of another security group in the same Region. To use
 this security group, specify \"self\". You must specify either AddressRanges parameter
 or SecurityGroupIds parameter, but you can also specify both.",
            "type": "array",
            "items": {
              "type": "string",
              "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$|^self$",
              "minLength": 1,
              "maxLength": 64
            }
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
```

```
            "Type",
            "Code",
            "AddressRanges",
            "SecurityGroupIds",
            "Description"
          ]
        },
        "required": [
          "Type",
          "Code"
        ]
      },
      "minItems": 0,
      "maxItems": 60,
      "uniqueItems": true
    },
    "Tags": {
      "description": "Up to 50 tags (key/value pairs) to categorize the security
group.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
```

```
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "VpcId",
      "SecurityGroupName",
      "SecurityGroupDescription",
      "TcpUdpIngressRules",
      "TcpUdpEgressRules",
      "IcmpIngressRules",
      "IcmpEgressRules",
      "Tags"
    ]
  },
  "required": [
    "VpcId",
    "SecurityGroupName",
    "SecurityGroupDescription"
  ]
}
```

Schema for Change Type ct-3pwbixz27n3tn

Classifications:

- Deployment | Managed landing zone | Management account | Create customer-managed application account

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create Customer-Managed Application Account",
  "description": "Create a customer-managed application account in a multi-account
 AWS landing zone. Customer-managed accounts give you full control to operate the
 infrastructure within the centralized architecture managed by AMS. Multi-account AWS
 landing zone core accounts must already be onboarded to AWS Managed Services (AMS).",
```

```
  "type": "object",
  "properties": {
    "AccountName": {
      "description": "A name for the new customer-managed application account. Max
 length 50 characters. The underscore (_) is not allowed.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9]{1}[a-zA-Z0-9.-]{0,49}$"
    },
    "AccountEmail": {
      "description": "The email address for the owner of the new customer-managed
 application account. The AccountEmail address must be unique per account.",
      "type": "string",
      "pattern": "^[a-zA-Z0-9_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$"
    },
    "CustomerManagedOUName": {
      "description": "The name of an existing customer-managed organizational unit (OU)
 for this account, in the form of <customer-managed ou name> or <customer-managed ou
 name>:<child ou name>. The default value is customer-managed. To create new OUs under
 customer-managed OU, please use create custom OU CT ct-1ksyoxreh35tu",
      "type": "string",
      "default": "customer-managed"
    }
  },
  "metadata": {
    "ui:order": [
      "AccountName",
      "AccountEmail",
      "CustomerManagedOUName"
    ]
  },
  "additionalProperties": false,
  "required": [
    "AccountName",
    "AccountEmail"
  ]
}
```

Schema for Change Type ct-3qe6io8t6jtny

Classifications:

- Management | AWS service | Self-provisioned service | Add (review required)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Self-Provisioned AWS Service",
  "description": "Add a specific, allowed, AWS service to your AMS account. AMS adds
 the necessary permissions to use the service to an existing IAM role that you specify,
 or creates a new role that allows you to use the service without AMS management under
 the AMS Shared Responsibility model. Compliance is a shared responsibility and your
 AMS compliance status does not automatically apply to services or applications that
 you add in this way. Some AWS services do not have compliance certifications. For
 more information, go to the AWS Services in Scope of AWS Assurance Program page.
 On that page, unless specifically excluded, features of each of the services are
 considered in scope of the assurance programs, and are reviewed and tested as part of
 the assessment.",
  "type": "object",
  "properties": {
    "ServiceName": {
      "description": "The name of the AWS service.",
      "type": "string",
      "enum": [
        "Alexa for Business",
        "Amazon API Gateway",
        "Amazon AppStream 2.0",
        "Amazon Athena",
        "Amazon CloudSearch",
        "Amazon CloudWatch Synthetics",
        "Amazon Cognito",
        "Amazon Comprehend",
        "Amazon Connect",
        "Amazon DocumentDB (with MongoDB compatibility)",
        "Amazon DynamoDB",
        "Amazon DevOps Guru",
        "Amazon ECR",
        "Amazon ECS on AWS Fargate",
        "Amazon EKS on AWS Fargate",
        "Amazon EMR",
        "Amazon EventBridge",
        "Amazon Forecast",
        "Amazon FSx",
        "Amazon Inspector",
        "Amazon Kinesis Data Analytics",
        "Amazon Kinesis Data Firehose",
        "Amazon Kinesis Data Streams",
        "Amazon Kinesis Video Streams",
```

```
        "Amazon Lex",
        "Amazon Managed Service for Prometheus",
        "Amazon Managed Streaming for Apache Kafka",
        "Amazon MQ",
        "Amazon Personalize",
        "Amazon QuickSight",
        "Amazon Rekognition",
        "Amazon SageMaker",
        "Amazon Simple Email Service",
        "Amazon Simple Workflow Service",
        "Amazon Textract",
        "Amazon Transcribe",
        "Amazon WorkDocs",
        "Amazon WorkSpaces",
        "AWS Amplify",
        "AWS Audit Manager",
        "AWS Batch",
        "AMS Code services",
        "AWS App Mesh",
        "AWS AppSync",
        "AWS Certificate Manager (ACM)",
        "AWS Private Certificate Authority (PCA)",
        "AWS CloudEndure",
        "AWS CloudHSM",
        "AWS CodeBuild",
        "AWS CodeCommit",
        "AWS CodeDeploy",
        "AWS CodePipeline",
        "AWS Compute Optimizer",
        "AWS DataSync",
        "AWS Elastic Disaster Recovery",
        "AWS Elemental MediaConvert",
        "AWS Elemental MediaLive",
        "AWS Elemental MediaPackage",
        "AWS Elemental MediaStore",
        "AWS Elemental MediaTailor",
        "AWS Global Accelerator",
        "AWS Glue",
        "AWS Lake Formation",
        "AWS Lambda",
        "AWS License Manager",
        "AWS Migration Hub",
        "AWS Outposts",
        "AWS Resilience Hub",
```

```
        "AWS Secrets Manager",
        "AWS Security Hub",
        "AWS Service Catalog AppRegistry",
        "AWS Shield",
        "AWS Snowball",
        "AWS Step Functions",
        "AWS Systems Manager Parameter Store",
        "AWS Systems Manager Automation",
        "AWS Transfer for SFTP",
        "AWS Transit Gateway",
        "AWS WAF - Web Application Firewall",
        "AWS Well Architected Tool",
        "AWS X-Ray",
        "EC2 Image Builder",
        "VM Import/Export"
      ]
    },
    "IAMRole": {
      "description": "ARN of an existing IAM role to add the permissions to self-
manage the AWS service. If left blank, a new role is created with the necessary
 permissions.",
      "type": "string",
      "pattern": "^arn:aws:iam::\\d{12}:role\\/[\\w+=,.@-]{1,64}$"
    },
    "SAMLProviders": {
      "description": "A single SAML provider name or a comma-separated list of SAML
 providers to use with the role",
      "type": "string",
      "pattern": "^[\\w+=,.@-]{1,256}$"
    },
    "Priority": {
      "description": "The priority of the request. See AMS \"RFC scheduling\"
 documentation for a definition of the priorities.",
      "type": "string",
      "enum": [
        "Low",
        "Medium",
        "High"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
```

```
      "ServiceName",
      "IAMRole",
      "SAMLProviders",
      "Priority"
    ]
  },
  "required": [
    "ServiceName"
  ]
}
```

Schema for Change Type ct-3r2ckznmt0a59

Classifications:

- Deployment | Managed landing zone | Networking account | Add static route

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Add Static Route",
  "description": "Create a static route on transit gateway (TGW) route table. Use this
 change type for multi-account landing zone (MALZ) Networking accounts only.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateRouteInTGWRouteTable.",
      "type": "string",
      "enum": [
        "AWSManagedServices-CreateRouteInTGWRouteTable"
      ],
      "default": "AWSManagedServices-CreateRouteInTGWRouteTable"
    },
    "Region": {
      "description": "The AWS Region in which the TGW attachment and TGW route table is
 located, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Blackhole": {
```

```
          "description": "True to indicate that the route's target isn't available. Do
 this when the traffic for the static route is to be dropped by the Transit Gateway.
 False to route the traffic to the specified TGW attachment ID. Default value is
 false.",
          "type": "array",
          "items": {
            "type": "boolean",
            "default": false
          },
          "minItems": 1,
          "maxItems": 1
        },
        "DestinationCidrBlock": {
          "description": "The IPV4 CIDR range used for destination matches. Routing
 decisions are based on the most specific match.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^(([0-9][0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([0-9]
[0-9]{0,1}|1[0-9]{2}|2[0-4][0-9]|25[0-5])(/([0-9]|[1-2][0-9]|3[0-2])){0,1}$"
          },
          "maxItems": 1
        },
        "TransitGatewayAttachmentId": {
          "description": "The TGW Attachment ID that will serve as route table target.
 If Blackhole is false, this parameter is required, otherwise leave this parameter
 blank.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-attach-[a-z0-9]{17}$"
          },
          "maxItems": 1
        },
        "TransitGatewayRouteTableId": {
          "description": "The ID of the TGW route table.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^tgw-rtb-[a-z0-9]{17}$"
          },
          "maxItems": 1
        }
      },
```

```
      "metadata": {
        "ui:order": [
          "TransitGatewayRouteTableId",
          "DestinationCidrBlock",
          "TransitGatewayAttachmentId",
          "Blackhole"
        ]
      },
      "additionalProperties": false,
      "required": [
        "TransitGatewayRouteTableId",
        "DestinationCidrBlock"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3rcl9u1k017wu

Classifications:

- Management | Monitoring and notification | SNS | Subscribe to DirectCustomerAlerts

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Subscribe to DirectCustomerAlerts",
  "description": "Subscribe an email address to the Direct-Customer-Alerts SNS
 topic.",
  "type": "object",
```

```
  "properties": {
    "Region": {
      "description": "The AWS Region of the account with the SNS subscription.",
      "type": "string",
      "pattern": "[a-z]{2}((-gov)|(-iso(b?)))?-[a-z]+-\\d{1}"
    },
    "Email": {
      "description": "The email address subscribing to the Direct-Customer-Alerts SNS
 topic.",
      "pattern": "^[a-zA-Z0-9.!#$%&'*+/=?^_`{|}~-]+@[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-
zA-Z0-9])?(?:\\.[a-zA-Z0-9](?:[a-zA-Z0-9-]{0,61}[a-zA-Z0-9])?)*$",
      "type": "string"
    }
  },
  "metadata": {
    "ui:order": [
      "Region",
      "Email"
    ]
  },
  "additionalProperties": false,
  "required": [
    "Region",
    "Email"
  ]
}
```

Schema for Change Type ct-3rd4781c2nnhp

Classifications:

- Management | Managed account | Direct Change mode | Enable

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Enable Direct Change mode",
  "description": "Enable Direct Change mode (DCM). DCM grants native AWS access to
 provision and update AWS resources. The resources and changes to them are fully
 supported by AMS, including monitoring, patch, backup, and incident response
 management.",
  "type": "object",
  "properties": {
```

```
    "SamlIdentityProviderArns": {
      "description": "Comma-separated list of ARNs (Amazon Resource Name) of the SAML
identity provider (IdP), or providers, to assume the DCM roles. You must set at least
one provider, using either this parameter, or one of the other provider parameters
(IamEntityArns or AwsServicePrincipals).",
      "type": "array",
      "items": {
        "type": "string"
      },
      "uniqueItems": true
    },
    "IamEntityArns": {
      "description": "Comma-separated list of ARNs of the IAM entities to assume the
DCM roles (example: role, user). You must set at least one provider, using either
this parameter, or one of the other provider parameters (SamlIdentityProviderArns or
AwsServicePrincipals).",
      "type": "array",
      "items": {
        "type": "string"
      },
      "uniqueItems": true
    },
    "AwsServicePrincipals": {
      "description": "Comma-separated list of AWS service principal names for a
service, or services, to assume the DCM roles (example: ecs.amazonaws.com). To
find a service principal name, see AWS documentation. You must set at least one
provider, using either this parameter, or one of the other provider parameters
(SamlIdentityProviderArns or IamEntityArns).",
      "type": "array",
      "items": {
        "type": "string"
      },
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "SamlIdentityProviderArns",
      "IamEntityArns",
      "AwsServicePrincipals"
    ]
  }
```

```
}
```

Schema for Change Type ct-3rk1nl1ufn5g3

Classifications:

- Management | AMS Resource Scheduler | Schedule | Delete

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Delete Resource Scheduler Schedule",
  "description": "Delete an existing schedule used in AMS Resource Scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-DeleteScheduleOrPeriod.",
      "type": "string",
      "enum": [
        "AWSManagedServices-DeleteScheduleOrPeriod"
      ],
      "default": "AWSManagedServices-DeleteScheduleOrPeriod"
    },
    "Region": {
      "description": "The AWS Region of the account where the AMS Resource Scheduler
 solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "ConfigurationType": {
          "description": "Specify the value: schedule. This explicitly requests that
 the Resource Scheduler schedule be deleted. The option cannot be left blank; it must
 be schedule.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "schedule"
            ],
            "default": "schedule"
```

```
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Name": {
          "description": "The name of the schedule to delete.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,64}$"
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "ConfigurationType",
          "Name"
        ]
      },
      "required": [
        "ConfigurationType",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3rqqu43krekby

Classifications:

- Deployment | Advanced stack components | AMI | Create


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create AMI",
  "description": "Create an Amazon Machine Image (AMI) based on an existing standalone
 EC2 instance in your AMS account. The instance must be in the stopped state before
 running this change type.",
  "type": "object",
  "properties": {
    "AmiName": {
      "description": "A name for the AMI. Must be unique per Region and account. If the
 name is not unique, the create AMI operation fails.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "InstanceId": {
      "description": "ID of the instance to create the AMI from, in the form of
 i-01234567890abcdef. The instance must be stopped. Specify a standalone EC2 instance,
 do not use an Auto Scaling group instance. Refer to the AMS User Guide documentation
 on creating AMIs for instructions on preparing the instance.",
      "type": "string",
      "pattern": "^i-[a-zA-Z0-9]{8}$|^i-[a-zA-Z0-9]{17}$"
    },
    "AmiTags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
```

```
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "InstanceId",
      "AmiName",
      "AmiTags"
    ]
  },
  "required": [
    "InstanceId",
    "AmiName"
  ]
}
```

Schema for Change Type ct-3s3ik03uzw19t

Classifications:

- Management | Advanced stack components | RDS database stack | Start DB instance

```
{
```

```
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Start RDS DB Instance",
  "description": "Start an Amazon Relational Database Service (RDS) database (DB)
 instance.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-StartRDSInstance.",
      "type": "string",
      "enum": [
        "AWSManagedServices-StartRDSInstance"
      ],
      "default": "AWSManagedServices-StartRDSInstance"
    },
    "Region": {
      "description": "The AWS Region in which the RDS DB is located, in the form us-
east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "InstanceId": {
          "description": "RDS DB instance identifier.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?=[a-zA-Z0-9-]{1,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstanceId"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstanceId"
      ]
    }
```

```
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3sk74t8igor0s

Classifications:

- Management | Advanced stack components | Target group | Attach instances

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Attach Instance Target To Target Group",
  "description": "Attach instance or instances to the target group (ALB and NLB).",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "AWSManagedServices-AttachInstancesToTargetGroup",
      "type": "string",
      "enum": [
        "AWSManagedServices-AttachInstancesToTargetGroup"
      ],
      "default": "AWSManagedServices-AttachInstancesToTargetGroup"
    },
    "Region": {
      "description": "The AWS Region where the target group and instances are located,
 in the form of us-east-1",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
```

```
    "Parameters": {
      "type": "object",
      "properties": {
        "InstancesIds": {
          "description": "The instance or instances IDs to be attached to the required
 target group, in the form of i-1234abcdef",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^i-[a-z0-9]{8,17}$"
          },
          "maxItems": 20
        },
        "InstancesPort": {
          "description": "The target instance port number.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^[0-9]{1,4}$|^[1-5][0-9]{4}$|^6[0-4][0-9]{3}$|^65[0-4][0-9]
{2}$|^655[0-2][0-9]$|^6553[0-5]$"
          },
          "maxItems": 1
        },
        "TargetGroupArn": {
          "description": "The target group Amazon Resource Name (ARN), in the
 form of arn:aws:elasticloadbalancing:us-west-2:123456789012:targetgroup/my-
targets/73e2d6bc24d8a067.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^arn:aws:elasticloadbalancing:([a-z]{2}((-gov))?-[a-z]+-\
\d{1}):[0-9]{0,12}:[a-zA-Z0-9\\_\\-\\/\\:]+$"
          },
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "InstancesIds",
          "InstancesPort",
          "TargetGroupArn"
        ]
      },
      "additionalProperties": false,
```

```
      "required": [
        "InstancesIds",
        "InstancesPort",
        "TargetGroupArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "additionalProperties": false,
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3skaisgnq0pf8

Classifications:

- Management | Advanced stack components | Identity and Access Management (IAM) | Update account alias

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update AWS Account Alias",
  "description": "Update an existing AWS account alias. Note that an AWS account can
 have only one alias. If you update the account alias, the new alias overwrites the
 previous alias, and the URL containing the previous alias stops working.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-CreateAccountAlias.",
      "type": "string",
      "enum": [
```

```
        "AWSManagedServices-CreateAccountAlias"
      ],
      "default": "AWSManagedServices-CreateAccountAlias"
    },
    "Region": {
      "description": "The AWS Region where the account is, in the form us-east-1.",
      "type": "string",
      "pattern": "[a-z]{2}-[a-z]+-\\d{1}"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "AWSAccountAlias": {
          "description": "The new alias name for the AWS account.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?=[a-zA-Z0-9-]{3,63}$)^[a-zA-Z][a-zA-Z0-9]*(-[a-zA-Z0-9]+)*$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "ReplaceAliasIfExists": {
          "description": "Specify True, to explicitly request that the current AWS
account alias name be updated. Must be True; cannot be left blank.",
          "type": "array",
          "items": {
            "enum": [
              "True"
            ],
            "type": "string",
            "default": "True"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "additionalProperties": false,
      "metadata": {
        "ui:order": [
          "AWSAccountAlias",
          "ReplaceAliasIfExists"
        ]
      },
```

```
      "required": [
        "AWSAccountAlias",
        "ReplaceAliasIfExists"
      ]
    }
  },
  "additionalProperties": false,
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ]
}
```

Schema for Change Type ct-3t4lifos8tu58

Classifications:

- Deployment | Advanced stack components | Target Group | Create (for NLB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create target group for NLB",
  "description": "Use to create a target group for a Network Load Balancer.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
```

```
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 0,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
```

```
      "description": "Must be stm-6pvp2f7cp481g1r47",
      "type": "string",
      "enum": [
        "stm-6pvp2f7cp481g1r47"
      ],
      "default": "stm-6pvp2f7cp481g1r47"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
 the change. This will not prolong execution, but the RFC fails if the change is not
 completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 60
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "NetworkLoadBalancerArn": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the network load balancer
 in the form arn:aws:elasticloadbalancing:region:account-id:loadbalancer/net/load-
balancer-name/load-balancer-id. This is used to create CloudWatch alarms that trigger
 if the Target Group contains no healthy instances.",
          "pattern": "arn:aws:elasticloadbalancing:[a-z1-9\\-]{9,15}:[0-9]
{12}:loadbalancer/net/[a-zA-Z0-9\\-]{1,32}/[a-z0-9]+"
        },
        "HealthCheckHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health check successes required to
 declare an EC2 instance healthy.",
          "pattern": "[2-9]{1}|10|^$",
          "default": "3"
        },
        "HealthCheckInterval": {
          "type": "integer",
          "description": "The approximate interval, in seconds, between health checks.
 The supported values are 10 or 30 seconds.",
          "default": 30
        },
        "HealthCheckTargetPath": {
          "type": "string",
```

```
          "description": "The ping path destination on the application hosts
 where the load balancer sends health check requests. Only applicable if
 HealthCheckTargetProtocol = HTTP or HTTPS.",
          "default": "/"
        },
        "HealthCheckTargetPort": {
          "type": "string",
          "description": "The port the load balancer uses when performing health checks
 on targets. The default is traffic-port, which indicates the port on which each target
 receives traffic from the load balancer.",
          "pattern": "[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|655[0-2]
[0-9]|6553[0-5]|traffic-port|",
          "default": ""
        },
        "HealthCheckTargetProtocol": {
          "type": "string",
          "description": "The protocol the load balancer uses when performing health
 checks on targets.",
          "enum": [
            "HTTP",
            "HTTPS",
            "TCP"
          ],
          "default": "TCP"
        },
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": "80"
        },
        "Name": {
          "type": "string",
          "description": "A name for the target group. This name must be unique per
 account, per region.",
          "pattern": "[0-9a-zA-Z\\-]{0,32}",
          "default": ""
        },
        "ProxyProtocolV2": {
          "type": "string",
          "description": "True if proxy protocol version 2 is enabled. False if it is
 not.",
```

```
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "DeregistrationDelayTimeout": {
          "type": "string",
          "description": "The amount of time, in seconds, for Elastic Load Balancing to
 wait before changing the state of a deregistering target from draining to unused.",
          "pattern": "(3600|3[0-5]{1}[0-9]{2}|[1-2]{1}[0-9]{3}|[0-9]{1,3})",
          "default": "300"
        },
        "TargetType": {
          "type": "string",
          "description": "The registration type of the targets; determines how you
 specify the TargetGroup targets. If you choose instance, you specify the targets by
 instance ID. If you choose ip, you specify the targets by IP address. After you create
 a target group, you cannot change its target type.",
          "enum": [
            "instance",
            "ip"
          ],
          "default": "instance"
        },
        "Target1ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target1Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target1AvailabilityZone": {
          "type": "string",
```

```
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target1ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target1ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target1ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        },
        "Target2ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target2Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target2AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target2ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target2ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target2ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        },
        "Target3ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
```

```
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target3Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target3AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target3ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target3ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target3ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|$",
          "default": ""
        },
        "Target4ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target4Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target4AvailabilityZone": {
          "type": "string",
```

```
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target4ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target4ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target4ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        },
        "Target5ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target5Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target5AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target5ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target5ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target5ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        },
        "Target6ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
```

```
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target6Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target6AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target6ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target6ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target6ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        },
        "Target7ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target7Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target7AvailabilityZone": {
          "type": "string",
```

```
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target7ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target7ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target7ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|$",
          "default": ""
        },
        "Target8ID": {
          "type": "string",
          "description": "ID of the EC2 instance to register a target, in the form
 i-0123abcd or i-01234567890abcdef if TargetType = instance. IP address if TargetType =
 ip. Leave blank if you don't need to register a target.",
          "pattern": "^$|i-[0-9a-f]{8}|i-[0-9a-f]{17}|((25[0-5]|2[0-4][0-9]|[01]?[0-9]
[0-9]?)(\\.|$)){4}",
          "default": ""
        },
        "Target8Port": {
          "type": "string",
          "description": "The port number on which the target is listening for
 traffic.",
          "pattern": "^$|[0-9]{1,4}|[1-5][0-9]{4}|6[0-4][0-9]{3}|65[0-4][0-9]{2}|
655[0-2][0-9]|6553[0-5]",
          "default": ""
        },
        "Target8AvailabilityZone": {
          "type": "string",
          "description": "Where the target receives traffic from. If the TargetType =
 ip, and the IP address in Target8ID is inside the VPC, leave blank. If the traffic is
 received from the specified AZ for the load balancer, and the TargetType = ip, and the
 IP address in Target8ID is outside the VPC, use the name of that AZ. If the traffic is
 received from all enabled AZs for the load balancer, and the TargetType = ip, and the
 IP address in Target8ID is outside the VPC, use all. If TargetType = instance, leave
 blank.",
          "pattern": "[a-z]{2,3}-[a-z\\-]{4,10}-[1-9]{1}[a-z]{1}|all|^$",
          "default": ""
        }
      },
      "metadata": {
        "ui:order": [
          "Name",
          "InstancePort",
```

```
          "NetworkLoadBalancerArn",
          "DeregistrationDelayTimeout",
          "ProxyProtocolV2",
          "HealthCheckTargetPath",
          "HealthCheckTargetPort",
          "HealthCheckTargetProtocol",
          "HealthCheckHealthyThreshold",
          "HealthCheckInterval",
          "TargetType",
          "Target1ID",
          "Target1Port",
          "Target1AvailabilityZone",
          "Target2ID",
          "Target2Port",
          "Target2AvailabilityZone",
          "Target3ID",
          "Target3Port",
          "Target3AvailabilityZone",
          "Target4ID",
          "Target4Port",
          "Target4AvailabilityZone",
          "Target5ID",
          "Target5Port",
          "Target5AvailabilityZone",
          "Target6ID",
          "Target6Port",
          "Target6AvailabilityZone",
          "Target7ID",
          "Target7Port",
          "Target7AvailabilityZone",
          "Target8ID",
          "Target8Port",
          "Target8AvailabilityZone"
        ]
      },
      "additionalProperties": false,
      "required": [
        "InstancePort",
        "NetworkLoadBalancerArn"
      ]
    }
  },
  "metadata": {
    "ui:order": [
```

```
      "Description",
      "VpcId",
      "Name",
      "Parameters",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "Parameters",
    "TimeoutInMinutes",
    "StackTemplateId"
  ],
  "additionalProperties": false
}
```

Schema for Change Type ct-3u61cd4edns0x

Classifications:

- Management | AMS Resource Scheduler | Schedule | Update

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Update Resource Scheduler Schedule",
  "description": "Update an existing schedule to be used in AMS Resource Scheduler.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-AddOrUpdateSchedule.",
      "type": "string",
      "enum": [
        "AWSManagedServices-AddOrUpdateSchedule"
      ],
      "default": "AWSManagedServices-AddOrUpdateSchedule"
    },
    "Region": {
```

```
      "description": "The AWS Region of the account where the AMS Resource Scheduler
solution is, in the form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "Action": {
          "description": "Specify the value: update. This explicitly requests that the
Resource Scheduler schedule be updated. The option cannot be left blank; it must be
update.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "update"
            ],
            "default": "update"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Name": {
          "description": "The name of the schedule to update.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,64}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Description": {
          "description": "A meaningful description for the schedule.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,1000}$|^$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Hibernate": {
```

```
        "description": "True to hibernate (suspend-to-disk) EC2 instances that are
enabled for hibernation and meet hibernation requirements, false to not. Check the
EC2 console to find out if your instances are enabled for hibernation. Default is
false.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        },
        "maxItems": 1,
        "minItems": 1
      },
      "Enforced": {
        "description": "True to enforce the schedule, false to not. When this field
is set to true, the Resource Scheduler will stop a running resource if it is manually
started outside of the running period, and it will start a resource if it is stopped
manually during the running period. Default is false.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "true",
            "false"
          ]
        },
        "maxItems": 1,
        "minItems": 1
      },
      "OverrideStatus": {
        "description": "Override the current schedule action. If set to running,
the instance will be started but not stopped until it is manually stopped. Similarly
when set to stopped, the instance will be stopped but not started automatically
until manually started. There is no default. If left unspecified this setting is not
used.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "running",
            "stopped"
          ]
```

```
          },
          "maxItems": 1,
          "minItems": 1
        },
        "Periods": {
          "description": "A comma-separated list of one or more period names in this
schedule. The name, or names, must match the existing defined periods.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "(?!^[-_, +=.:#/@])^[A-Za-z0-9-_, +=.:#/@]{1,2000}$"
          },
          "maxItems": 1,
          "minItems": 1
        },
        "RetainRunning": {
          "description": "True to prevent the Resource Scheduler from stopping a
resource at the end of a period if the instance was manually started before the
beginning of the period. False to not. Default is false.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "StopNewInstances": {
          "description": "True to stop a resource the first time it is tagged if it is
running outside of the running period. False to not stop the resource. The default is
true.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
```

```
      },
      "SSMMaintenanceWindow": {
        "description": "Comma-separated name or names of one, or more, existing AWS
Systems Manager maintenance windows, to use as the period. First, ensure that the
UseMaintenanceWindow parameter is set to true. Create a maintenance window with the
Deployment | Patching | SSM patch window | Create change type (ct-0el2j07llrxs7).",
        "type": "array",
        "items": {
          "type": "string",
          "pattern": "(?!^[-_, ]$)^[A-Za-z0-9-_, ]{1,4096}$|^$"
        },
        "maxItems": 1,
        "minItems": 1
      },
      "TimeZone": {
        "description": "The name of the time zone, in the form US/Pacific, the
schedule uses. If no time zone is specified then the time zone DefaultTimezone set
when the Resource Scheduler was deployed is used.",
        "type": "array",
        "items": {
          "type": "string",
          "enum": [
            "Africa/Abidjan",
            "Africa/Accra",
            "Africa/Addis_Ababa",
            "Africa/Algiers",
            "Africa/Asmara",
            "Africa/Bamako",
            "Africa/Bangui",
            "Africa/Banjul",
            "Africa/Bissau",
            "Africa/Blantyre",
            "Africa/Brazzaville",
            "Africa/Bujumbura",
            "Africa/Cairo",
            "Africa/Casablanca",
            "Africa/Ceuta",
            "Africa/Conakry",
            "Africa/Dakar",
            "Africa/Dar_es_Salaam",
            "Africa/Djibouti",
            "Africa/Douala",
            "Africa/El_Aaiun",
            "Africa/Freetown",
```

```
                "Africa/Gaborone",
                "Africa/Harare",
                "Africa/Johannesburg",
                "Africa/Juba",
                "Africa/Kampala",
                "Africa/Khartoum",
                "Africa/Kigali",
                "Africa/Kinshasa",
                "Africa/Lagos",
                "Africa/Libreville",
                "Africa/Lome",
                "Africa/Luanda",
                "Africa/Lubumbashi",
                "Africa/Lusaka",
                "Africa/Malabo",
                "Africa/Maputo",
                "Africa/Maseru",
                "Africa/Mbabane",
                "Africa/Mogadishu",
                "Africa/Monrovia",
                "Africa/Nairobi",
                "Africa/Ndjamena",
                "Africa/Niamey",
                "Africa/Nouakchott",
                "Africa/Ouagadougou",
                "Africa/Porto-Novo",
                "Africa/Sao_Tome",
                "Africa/Tripoli",
                "Africa/Tunis",
                "Africa/Windhoek",
                "America/Adak",
                "America/Anchorage",
                "America/Anguilla",
                "America/Antigua",
                "America/Araguaina",
                "America/Argentina/Buenos_Aires",
                "America/Argentina/Catamarca",
                "America/Argentina/Cordoba",
                "America/Argentina/Jujuy",
                "America/Argentina/La_Rioja",
                "America/Argentina/Mendoza",
                "America/Argentina/Rio_Gallegos",
                "America/Argentina/Salta",
                "America/Argentina/San_Juan",
```

```
                "America/Argentina/San_Luis",
                "America/Argentina/Tucuman",
                "America/Argentina/Ushuaia",
                "America/Aruba",
                "America/Asuncion",
                "America/Atikokan",
                "America/Bahia",
                "America/Bahia_Banderas",
                "America/Barbados",
                "America/Belem",
                "America/Belize",
                "America/Blanc-Sablon",
                "America/Boa_Vista",
                "America/Bogota",
                "America/Boise",
                "America/Cambridge_Bay",
                "America/Campo_Grande",
                "America/Cancun",
                "America/Caracas",
                "America/Cayenne",
                "America/Cayman",
                "America/Chicago",
                "America/Chihuahua",
                "America/Costa_Rica",
                "America/Creston",
                "America/Cuiaba",
                "America/Curacao",
                "America/Danmarkshavn",
                "America/Dawson",
                "America/Dawson_Creek",
                "America/Denver",
                "America/Detroit",
                "America/Dominica",
                "America/Edmonton",
                "America/Eirunepe",
                "America/El_Salvador",
                "America/Fortaleza",
                "America/Glace_Bay",
                "America/Godthab",
                "America/Goose_Bay",
                "America/Grand_Turk",
                "America/Grenada",
                "America/Guadeloupe",
                "America/Guatemala",
```

```
                "America/Guayaquil",
                "America/Guyana",
                "America/Halifax",
                "America/Havana",
                "America/Hermosillo",
                "America/Indiana/Indianapolis",
                "America/Indiana/Knox",
                "America/Indiana/Marengo",
                "America/Indiana/Petersburg",
                "America/Indiana/Tell_City",
                "America/Indiana/Vevay",
                "America/Indiana/Vincennes",
                "America/Indiana/Winamac",
                "America/Inuvik",
                "America/Iqaluit",
                "America/Jamaica",
                "America/Juneau",
                "America/Kentucky/Louisville",
                "America/Kentucky/Monticello",
                "America/Kralendijk",
                "America/La_Paz",
                "America/Lima",
                "America/Los_Angeles",
                "America/Lower_Princes",
                "America/Maceio",
                "America/Managua",
                "America/Manaus",
                "America/Marigot",
                "America/Martinique",
                "America/Matamoros",
                "America/Mazatlan",
                "America/Menominee",
                "America/Merida",
                "America/Metlakatla",
                "America/Mexico_City",
                "America/Miquelon",
                "America/Moncton",
                "America/Monterrey",
                "America/Montevideo",
                "America/Montreal",
                "America/Montserrat",
                "America/Nassau",
                "America/New_York",
                "America/Nipigon",
```

```
                "America/Nome",
                "America/Noronha",
                "America/North_Dakota/Beulah",
                "America/North_Dakota/Center",
                "America/North_Dakota/New_Salem",
                "America/Ojinaga",
                "America/Panama",
                "America/Pangnirtung",
                "America/Paramaribo",
                "America/Phoenix",
                "America/Port-au-Prince",
                "America/Port_of_Spain",
                "America/Porto_Velho",
                "America/Puerto_Rico",
                "America/Rainy_River",
                "America/Rankin_Inlet",
                "America/Recife",
                "America/Regina",
                "America/Resolute",
                "America/Rio_Branco",
                "America/Santa_Isabel",
                "America/Santarem",
                "America/Santiago",
                "America/Santo_Domingo",
                "America/Sao_Paulo",
                "America/Scoresbysund",
                "America/Sitka",
                "America/St_Barthelemy",
                "America/St_Johns",
                "America/St_Kitts",
                "America/St_Lucia",
                "America/St_Thomas",
                "America/St_Vincent",
                "America/Swift_Current",
                "America/Tegucigalpa",
                "America/Thule",
                "America/Thunder_Bay",
                "America/Tijuana",
                "America/Toronto",
                "America/Tortola",
                "America/Vancouver",
                "America/Whitehorse",
                "America/Winnipeg",
                "America/Yakutat",
```

```
                "America/Yellowknife",
                "Antarctica/Casey",
                "Antarctica/Davis",
                "Antarctica/DumontDUrville",
                "Antarctica/Macquarie",
                "Antarctica/Mawson",
                "Antarctica/McMurdo",
                "Antarctica/Palmer",
                "Antarctica/Rothera",
                "Antarctica/Syowa",
                "Antarctica/Vostok",
                "Arctic/Longyearbyen",
                "Asia/Aden",
                "Asia/Almaty",
                "Asia/Amman",
                "Asia/Anadyr",
                "Asia/Aqtau",
                "Asia/Aqtobe",
                "Asia/Ashgabat",
                "Asia/Baghdad",
                "Asia/Bahrain",
                "Asia/Baku",
                "Asia/Bangkok",
                "Asia/Beirut",
                "Asia/Bishkek",
                "Asia/Brunei",
                "Asia/Choibalsan",
                "Asia/Chongqing",
                "Asia/Colombo",
                "Asia/Damascus",
                "Asia/Dhaka",
                "Asia/Dili",
                "Asia/Dubai",
                "Asia/Dushanbe",
                "Asia/Gaza",
                "Asia/Harbin",
                "Asia/Hebron",
                "Asia/Ho_Chi_Minh",
                "Asia/Hong_Kong",
                "Asia/Hovd",
                "Asia/Irkutsk",
                "Asia/Jakarta",
                "Asia/Jayapura",
                "Asia/Jerusalem",
```

```
                "Asia/Kabul",
                "Asia/Kamchatka",
                "Asia/Karachi",
                "Asia/Kashgar",
                "Asia/Kathmandu",
                "Asia/Khandyga",
                "Asia/Kolkata",
                "Asia/Krasnoyarsk",
                "Asia/Kuala_Lumpur",
                "Asia/Kuching",
                "Asia/Kuwait",
                "Asia/Macau",
                "Asia/Magadan",
                "Asia/Makassar",
                "Asia/Manila",
                "Asia/Muscat",
                "Asia/Nicosia",
                "Asia/Novokuznetsk",
                "Asia/Novosibirsk",
                "Asia/Omsk",
                "Asia/Oral",
                "Asia/Phnom_Penh",
                "Asia/Pontianak",
                "Asia/Pyongyang",
                "Asia/Qatar",
                "Asia/Qyzylorda",
                "Asia/Rangoon",
                "Asia/Riyadh",
                "Asia/Sakhalin",
                "Asia/Samarkand",
                "Asia/Seoul",
                "Asia/Shanghai",
                "Asia/Singapore",
                "Asia/Taipei",
                "Asia/Tashkent",
                "Asia/Tbilisi",
                "Asia/Tehran",
                "Asia/Thimphu",
                "Asia/Tokyo",
                "Asia/Ulaanbaatar",
                "Asia/Urumqi",
                "Asia/Ust-Nera",
                "Asia/Vientiane",
                "Asia/Vladivostok",
```

```
                "Asia/Yakutsk",
                "Asia/Yekaterinburg",
                "Asia/Yerevan",
                "Atlantic/Azores",
                "Atlantic/Bermuda",
                "Atlantic/Canary",
                "Atlantic/Cape_Verde",
                "Atlantic/Faroe",
                "Atlantic/Madeira",
                "Atlantic/Reykjavik",
                "Atlantic/South_Georgia",
                "Atlantic/St_Helena",
                "Atlantic/Stanley",
                "Australia/Adelaide",
                "Australia/Brisbane",
                "Australia/Broken_Hill",
                "Australia/Currie",
                "Australia/Darwin",
                "Australia/Eucla",
                "Australia/Hobart",
                "Australia/Lindeman",
                "Australia/Lord_Howe",
                "Australia/Melbourne",
                "Australia/Perth",
                "Australia/Sydney",
                "Canada/Atlantic",
                "Canada/Central",
                "Canada/Eastern",
                "Canada/Mountain",
                "Canada/Newfoundland",
                "Canada/Pacific",
                "Europe/Amsterdam",
                "Europe/Andorra",
                "Europe/Athens",
                "Europe/Belgrade",
                "Europe/Berlin",
                "Europe/Bratislava",
                "Europe/Brussels",
                "Europe/Bucharest",
                "Europe/Budapest",
                "Europe/Busingen",
                "Europe/Chisinau",
                "Europe/Copenhagen",
                "Europe/Dublin",
```

```
            "Europe/Gibraltar",
            "Europe/Guernsey",
            "Europe/Helsinki",
            "Europe/Isle_of_Man",
            "Europe/Istanbul",
            "Europe/Jersey",
            "Europe/Kaliningrad",
            "Europe/Kiev",
            "Europe/Lisbon",
            "Europe/Ljubljana",
            "Europe/London",
            "Europe/Luxembourg",
            "Europe/Madrid",
            "Europe/Malta",
            "Europe/Mariehamn",
            "Europe/Minsk",
            "Europe/Monaco",
            "Europe/Moscow",
            "Europe/Oslo",
            "Europe/Paris",
            "Europe/Podgorica",
            "Europe/Prague",
            "Europe/Riga",
            "Europe/Rome",
            "Europe/Samara",
            "Europe/San_Marino",
            "Europe/Sarajevo",
            "Europe/Simferopol",
            "Europe/Skopje",
            "Europe/Sofia",
            "Europe/Stockholm",
            "Europe/Tallinn",
            "Europe/Tirane",
            "Europe/Uzhgorod",
            "Europe/Vaduz",
            "Europe/Vatican",
            "Europe/Vienna",
            "Europe/Vilnius",
            "Europe/Volgograd",
            "Europe/Warsaw",
            "Europe/Zagreb",
            "Europe/Zaporozhye",
            "Europe/Zurich",
            "GMT",
```

```
                "Indian/Antananarivo",
                "Indian/Chagos",
                "Indian/Christmas",
                "Indian/Cocos",
                "Indian/Comoro",
                "Indian/Kerguelen",
                "Indian/Mahe",
                "Indian/Maldives",
                "Indian/Mauritius",
                "Indian/Mayotte",
                "Indian/Reunion",
                "Pacific/Apia",
                "Pacific/Auckland",
                "Pacific/Chatham",
                "Pacific/Chuuk",
                "Pacific/Easter",
                "Pacific/Efate",
                "Pacific/Enderbury",
                "Pacific/Fakaofo",
                "Pacific/Fiji",
                "Pacific/Funafuti",
                "Pacific/Galapagos",
                "Pacific/Gambier",
                "Pacific/Guadalcanal",
                "Pacific/Guam",
                "Pacific/Honolulu",
                "Pacific/Johnston",
                "Pacific/Kiritimati",
                "Pacific/Kosrae",
                "Pacific/Kwajalein",
                "Pacific/Majuro",
                "Pacific/Marquesas",
                "Pacific/Midway",
                "Pacific/Nauru",
                "Pacific/Niue",
                "Pacific/Norfolk",
                "Pacific/Noumea",
                "Pacific/Pago_Pago",
                "Pacific/Palau",
                "Pacific/Pitcairn",
                "Pacific/Pohnpei",
                "Pacific/Port_Moresby",
                "Pacific/Rarotonga",
                "Pacific/Saipan",
```

```
              "Pacific/Tahiti",
              "Pacific/Tarawa",
              "Pacific/Tongatapu",
              "Pacific/Wake",
              "Pacific/Wallis",
              "US/Alaska",
              "US/Arizona",
              "US/Central",
              "US/Eastern",
              "US/Hawaii",
              "US/Mountain",
              "US/Pacific",
              "UTC"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "UseMaintenanceWindow": {
          "description": "True to add an Amazon RDS maintenance window as a period to
an Amazon RDS instance schedule, or to add an AWS Systems Manager (SSM) maintenance
window as a period to an Amazon EC2 instance schedule. An RDS maintenance window is
automatically created by RDS. An SSM maintenance window you create with the Deployment
| Patching | SSM maintenance window | Create (ct-0el2j07llrxs7) change type. False
to not add either maintenance window, but to use the start and stop settings of the
period.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        },
        "UseMetrics": {
          "description": "Enable CloudWatch metrics for this schedule. This field
overrides the default settings defined when the Resource Scheduler was deployed.",
          "type": "array",
          "items": {
            "type": "string",
            "enum": [
```

```
              "true",
              "false"
            ]
          },
          "maxItems": 1,
          "minItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "Action",
          "Name",
          "Description",
          "Hibernate",
          "Enforced",
          "OverrideStatus",
          "Periods",
          "RetainRunning",
          "StopNewInstances",
          "SSMMaintenanceWindow",
          "TimeZone",
          "UseMaintenanceWindow",
          "UseMetrics"
        ]
      },
      "required": [
        "Action",
        "Name"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
```

```
  "additionalProperties": false
}
```

Schema for Change Type ct-3u9yd8jznb2zd

Classifications:

- Management | Advanced stack components | AMI | Encrypt

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Encrypt AMI",
  "description": "Use to create a custom AMI with an encrypted EBS snapshot, which
 protects data at rest. When the encrypted AMI is launched, the corresponding EBS
 volume is encrypted.",
  "type": "object",
  "properties": {
    "AmiId": {
      "description": "ID of the AMI to encrypt, in the form ami-0123abcd or
 ami-01234567890abcdef. The new AMI appends a date/time stamp and 'encrypted-copy' to
 the name of the source AMI.",
      "type": "string",
      "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$"
    },
    "KmsKeyId": {
      "description": "A KMS key to encrypt the AMI with. If one is not specified,
 the default EBS KMS key for the account is used. Allows any format specified in
 EC2 documentation for CopyImage API: https://docs.aws.amazon.com/AWSEC2/latest/
APIReference/API_CopyImage.html",
      "type": "string",
      "metadata": {
        "ams:sensitive": true
      }
    },
    "VpcId": {
      "description": "ID of the VPC where the source AMI is available and where the
 encrypted AMI will be created, in the form vpc-0123abcd or vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    }
  },
  "additionalProperties": false,
```

```
  "required": [
    "AmiId",
    "VpcId"
  ]
}
```

Schema for Change Type ct-3vfxkiudtovm9

Classifications:

- Management | Patching | Patch window | Set status

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Set Patch Window Status",
  "description": "Enable or disable an existing AWS Systems Manager (SSM) patch window.
 If the window is enabled, any task associated with it runs on the next occurrence
 of the maintenance window. If disabled, any future occurrences of the window no
 longer run. Occurrences of the window that are already running continue to run until
 completion.",
  "type": "object",
  "properties": {
    "DocumentName": {
      "description": "Must be AWSManagedServices-SetSsmMaintenanceWindowStatus.",
      "type": "string",
      "enum": [
        "AWSManagedServices-SetSsmMaintenanceWindowStatus"
      ],
      "default": "AWSManagedServices-SetSsmMaintenanceWindowStatus"
    },
    "Region": {
      "description": "The AWS Region in which the maintenance window is located, in the
 form us-east-1.",
      "type": "string",
      "pattern": "^([a-z]{2}((-gov))?-[a-z]+-\\d{1})$"
    },
    "Parameters": {
      "type": "object",
      "properties": {
        "MaintenanceWindowId": {
          "description": "The ID of the SSM patch maintenance window to set the status
 for.",
```

```
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^mw-[0-9a-f]{17}$"
          },
          "minItems": 1,
          "maxItems": 1
        },
        "Enabled": {
          "description": "True to enable the patch window, false to disable it.",
          "type": "array",
          "items": {
            "type": "boolean"
          },
          "minItems": 1,
          "maxItems": 1
        }
      },
      "metadata": {
        "ui:order": [
          "MaintenanceWindowId",
          "Enabled"
        ]
      },
      "required": [
        "MaintenanceWindowId",
        "Enabled"
      ],
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "DocumentName",
      "Region",
      "Parameters"
    ]
  },
  "required": [
    "DocumentName",
    "Region",
    "Parameters"
  ],
  "additionalProperties": false
```

```
}
```

Schema for Change Type ct-3w4lxdl3pqxob

Classifications:

- Deployment | Standard stacks | High availability one-tier stack | Create (with ELB)

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "name": "Create HA One-Tier Stack With ELB",
  "description": "Create a stack with an Auto Scaling Group, and an Elastic Load
 Balancer (ELB) with up to two listeners, integrated with an existing security group
 that you specify.",
  "type": "object",
  "properties": {
    "Description": {
      "description": "Meaningful information about the resource to be created.",
      "type": "string",
      "minLength": 1,
      "maxLength": 500
    },
    "VpcId": {
      "description": "ID of the VPC to use, in the form vpc-0123abcd or
 vpc-01234567890abcdef.",
      "type": "string",
      "pattern": "^vpc-[a-z0-9]{8}$|^vpc-[a-z0-9]{17}$"
    },
    "Name": {
      "description": "A name for the stack or stack component; this becomes the Stack
 Name.",
      "type": "string",
      "minLength": 1,
      "maxLength": 255
    },
    "Tags": {
      "description": "Up to fifty tags (key/value pairs) to categorize the resource.",
      "type": "array",
      "items": {
        "type": "object",
        "properties": {
          "Key": {
```

```
            "type": "string",
            "minLength": 1,
            "maxLength": 127
          },
          "Value": {
            "type": "string",
            "minLength": 1,
            "maxLength": 255
          }
        },
        "additionalProperties": false,
        "metadata": {
          "ui:order": [
            "Key",
            "Value"
          ]
        },
        "required": [
          "Key",
          "Value"
        ]
      },
      "minItems": 1,
      "maxItems": 50,
      "uniqueItems": true
    },
    "StackTemplateId": {
      "description": "Must be stm-g7rc538l62r4c23nb",
      "type": "string",
      "enum": [
        "stm-g7rc538l62r4c23nb"
      ],
      "default": "stm-g7rc538l62r4c23nb"
    },
    "TimeoutInMinutes": {
      "description": "The maximum amount of time, in minutes, to allow for execution of
the change. This will not prolong execution, but the RFC fails if the change is not
completed in the specified time.",
      "type": "number",
      "minimum": 0,
      "maximum": 360,
      "default": 360
    },
    "AutoScaling": {
```

```
      "type": "object",
      "properties": {
        "AmiId": {
          "type": "string",
          "description": "ID of the AMI for the Auto Scaling group to use when creating
new instances, in the form ami-0123abcd or ami-01234567890abcdef.",
          "pattern": "^ami-[a-zA-Z0-9]{8}$|^ami-[a-zA-Z0-9]{17}$"
        },
        "Cooldown": {
          "type": "string",
          "description": "The number of seconds after a scaling activity is completed
before any further scaling activities can start.",
          "default": "300"
        },
        "EBSOptimized": {
          "type": "string",
          "description": "True to create EBS-optimized instances, false to not.
EBS-optimization provides dedicated throughput to Amazon EBS and optimal EBS I/O
performance.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "HealthCheckGracePeriod": {
          "type": "string",
          "description": "The amount of time, in seconds, that auto scaling waits
before checking the health status of an EC2 instance that has come into service.
During this time, any health check failures for the instance are ignored.",
          "default": "600"
        },
        "HealthCheckType": {
          "type": "string",
          "description": "The service from which the health status is used, Amazon EC2
or Elastic Load Balancer.",
          "enum": [
            "EC2",
            "ELB"
          ],
          "default": "EC2"
        },
        "IAMInstanceProfile": {
          "type": "string",
```

```
          "description": "The IAM instance profile for the Auto Scaling group. EC2
 instances launched with an IAM role automatically have AWS security credentials
 available.",
          "pattern": "^customer[\\w-]+$",
          "default": "customer-mc-ec2-instance-profile"
        },
        "DetailedMonitoring": {
          "type": "string",
          "description": "True to enable detailed monitoring on the instances in the
 Auto Scaling group, false to use only basic monitoring.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "RootVolumeIops": {
          "type": "string",
          "description": "The IOPS to use for the root volume if volume type is io1,
 io2, or gp3."
        },
        "RootVolumeName": {
          "type": "string",
          "description": "The device name of the root volume (/dev/xvda or /dev/
sda1).",
          "enum": [
            "/dev/xvda",
            "/dev/sda1"
          ],
          "default": "/dev/xvda"
        },
        "RootVolumeSize": {
          "type": "integer",
          "description": "The size of the root volume for the instance in GiB.",
          "minimum": 8,
          "maximum": 16000
        },
        "RootVolumeThroughput": {
          "type": "integer",
          "description": "The throughput in MiB/s to provision for the root volume if
 the volume type is gp3.",
          "minimum": 125,
          "maximum": 1000
        },
```

```
        "RootVolumeType": {
          "type": "string",
          "description": "The type of the root volume for the instance. The default is
gp3.",
          "enum": [
            "standard",
            "io1",
            "io2",
            "gp2",
            "gp3"
          ],
          "default": "gp3"
        },
        "InstanceType": {
          "type": "string",
          "description": "The instance type for the Auto Scaling group to use when
creating new EC2 instances.",
          "default": "m4.large"
        },
        "MaxBatchSize": {
          "type": "integer",
          "description": "The maximum number of Auto Scaling group instances that AWS
CloudFormation updates at a time.",
          "default": 1
        },
        "MaxInstances": {
          "type": "string",
          "description": "The maximum number of instances you want in the Auto Scaling
group at any time.",
          "default": "2"
        },
        "MinInstances": {
          "type": "string",
          "description": "The minimum number of instances you want in the Auto Scaling
group at any time.",
          "default": "2"
        },
        "MinInstancesInService": {
          "type": "integer",
          "description": "The minimum number of instances that you want in service
within the Auto Scaling group while AWS CloudFormation updates old instances.",
          "default": 1
        },
        "ScaleDownPolicyCooldown": {
```

```
          "type": "string",
          "description": "The number of seconds after a scale-down activity is
completed before any further scaling activities can start.",
          "default": "300"
        },
        "ScaleDownPolicyEvaluationPeriods": {
          "type": "string",
          "description": "The number of periods over which data is compared to the
specified ScaleMetricName threshold.",
          "default": "4"
        },
        "ScaleDownPolicyPeriod": {
          "type": "string",
          "description": "The time over which the specified ScaleDownPolicyStatistic is
applied. You must specify a time in seconds that is a multiple of 60.",
          "default": "60"
        },
        "ScaleDownPolicyScalingAdjustment": {
          "type": "string",
          "description": "The number of instances by which to scale down.",
          "default": "-1"
        },
        "ScaleDownPolicyStatistic": {
          "type": "string",
          "description": "The statistic to apply to the scaling down alarm's associated
metric (ScaleMetricName).",
          "enum": [
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ],
          "default": "Average"
        },
        "ScaleDownPolicyThreshold": {
          "type": "string",
          "description": "The value against which the specified
ScaleDownPolicyStatistic is compared.",
          "default": "35"
        },
        "ScaleMetricName": {
          "type": "string",
          "description": "The metric to use in a scaling event.",
```

```
          "enum": [
            "CPUCreditUsage",
            "CPUCreditBalance",
            "CPUUtilization",
            "DiskReadOps",
            "DiskWriteOps",
            "DiskReadBytes",
            "DiskWriteBytes",
            "NetworkIn",
            "NetworkOut",
            "StatusCheckFailed",
            "StatusCheckFailed_Instance",
            "StatusCheckFailed_System"
          ],
          "default": "CPUUtilization"
        },
        "ScaleUpPolicyCooldown": {
          "type": "string",
          "description": "The number of seconds after a scale-up activity is completed
before any further scaling activities can start.",
          "default": "60"
        },
        "ScaleUpPolicyEvaluationPeriods": {
          "type": "string",
          "description": "The number of periods over which data is compared to the
specified ScaleMetricName threshold.",
          "default": "2"
        },
        "ScaleUpPolicyPeriod": {
          "type": "string",
          "description": "The time over which ScaleUpPolicyStatistic is applied. You
must specify a time in seconds that is a multiple of 60.",
          "default": "60"
        },
        "ScaleUpPolicyScalingAdjustment": {
          "type": "string",
          "description": "The number of instances by which to scale up.",
          "default": "2"
        },
        "ScaleUpPolicyStatistic": {
          "type": "string",
          "description": "The statistic to apply to the scaling up alarm's associated
metric (ScaleMetricName).",
          "enum": [
```

```
            "SampleCount",
            "Average",
            "Sum",
            "Minimum",
            "Maximum"
          ],
          "default": "Average"
        },
        "ScaleUpPolicyThreshold": {
          "type": "string",
          "description": "The value against which the specified ScaleUpPolicyStatistic
is compared.",
          "default": "75"
        },
        "SubnetIds": {
          "description": "One or more subnets for the Auto Scaling group to launch
instances into (scale up) or remove instances from (scale down), in the form
subnet-0123abcd or subnet-01234567890abcdef.",
          "type": "array",
          "items": {
            "type": "string",
            "pattern": "^subnet-([a-z0-9]{17}|[a-z0-9]{8})$"
          },
          "uniqueItems": true
        },
        "UserData": {
          "type": "array",
          "description": "A comma-delimited list where each element is a line of script
to be run on boot.",
          "items": {
            "type": "string"
          },
          "uniqueItems": true
        }
      },
      "metadata": {
        "ui:order": [
          "AmiId",
          "InstanceType",
          "RootVolumeIops",
          "RootVolumeName",
          "RootVolumeSize",
          "RootVolumeThroughput",
          "RootVolumeType",
```

```
          "EBSOptimized",
          "MaxInstances",
          "MinInstances",
          "IAMInstanceProfile",
          "SubnetIds",
          "UserData",
          "MaxBatchSize",
          "MinInstancesInService",
          "HealthCheckType",
          "HealthCheckGracePeriod",
          "DetailedMonitoring",
          "Cooldown",
          "ScaleMetricName",
          "ScaleUpPolicyCooldown",
          "ScaleUpPolicyEvaluationPeriods",
          "ScaleUpPolicyPeriod",
          "ScaleUpPolicyScalingAdjustment",
          "ScaleUpPolicyStatistic",
          "ScaleUpPolicyThreshold",
          "ScaleDownPolicyCooldown",
          "ScaleDownPolicyEvaluationPeriods",
          "ScaleDownPolicyPeriod",
          "ScaleDownPolicyScalingAdjustment",
          "ScaleDownPolicyStatistic",
          "ScaleDownPolicyThreshold"
        ]
      },
      "required": [
        "AmiId",
        "SubnetIds"
      ],
      "additionalProperties": false
    },
    "LoadBalancer": {
      "type": "object",
      "properties": {
        "Name": {
          "type": "string",
          "description": "A friendly name for the load balancer.",
          "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,31}$|^$"
        },
        "Public": {
          "type": "string",
```

```
          "description": "True if the load balancer endpoint is public, false if it is
private.",
          "enum": [
            "true",
            "false"
          ],
          "default": "false"
        },
        "SecurityGroups": {
          "type": "string",
          "description": "A list of security groups to associate with the load
balancer.",
          "pattern": "^sg-[a-z0-9]{8}$|^sg-[a-z0-9]{17}$"
        },
        "SubnetIds": {
          "type": "array",
          "description": "A list of subnet IDs that the Elastic Load Balancing creates
load balancer nodes in. For an Internet-facing load balancer provide a public subnet
ID, for an internal load balancer we recommend private subnet IDs.",
          "items": {
            "type": "string",
            "pattern": "^subnet-([a-z0-9]{17}|[a-z0-9]{8})$"
          },
          "uniqueItems": true
        },
        "AccessLogInterval": {
          "type": "string",
          "description": "The time interval, in minutes, to upload the load balancer
access log to the specified S3 bucket. Defaults to 60 Minutes.",
          "enum": [
            "5",
            "60"
          ],
          "default": "60"
        },
        "ConnectionDrainingTimeout": {
          "type": "integer",
          "description": "The maximum time, in seconds, to keep the existing
connections open before deregistering the instances.",
          "default": 60,
          "minimum": 1,
          "maximum": 3600
        },
        "IdleTimeout": {
```

```
          "type": "integer",
          "description": "The time, in seconds, that a connection to the load balancer
 can remain idle (no data is sent over the connection). After the specified time, the
 load balancer closes the connection.",
          "default": 60,
          "minimum": 1,
          "maximum": 3600
        },
        "CrossZone": {
          "type": "string",
          "description": "True to enable cross-zone load balancing (the load balancer
 nodes route traffic to the back-end instances across all Availability Zones), false to
 disable. Default is true.",
          "enum": [
            "true",
            "false"
          ],
          "default": "true"
        },
        "HealthCheckHealthyThreshold": {
          "type": "string",
          "description": "The number of consecutive health probe successes required
 before moving the instance to the healthy state after it was moved to unhealthy.",
          "pattern": "^[1-9]{1}[0-9]{0,1}$",
          "default": "2"
        },
        "HealthCheckInterval": {
          "type": "string",
          "description": "How often, in seconds, that health checks are run on an
 individual load balancer node.",
          "pattern": "^[1-9]{1}[0-9]{0,3}$",
          "default": "10"
        },
        "HealthCheckTarget": {
          "type": "string",
          "description": "The protocol, port, and path of the instance to check. The
 protocol can be TCP, HTTP, HTTPS, or SSL and valid ports are 1 through 65535. For TCP/
SSL no path is required. For HTTP/HTTPS, you must include a ping path in the string.
 For example, HTTP:80/weather/us/wa/seattle.",
          "pattern": "^(HTTP|HTTPS):[0-9]{1,5}[/][\\w./-]*$|^(SSL|TCP):[0-9]{1,5}$",
          "default": "TCP:80"
        },
        "HealthCheckTimeout": {
          "type": "string",
```

```
        "description": "The amount of time, in seconds, during which no response
means a failed health probe. This value must be less than the value for Interval.",
        "pattern": "^[1-9]{1}[0-9]{0,3}$",
        "default": "5"
      },
      "HealthCheckUnhealthyThreshold": {
        "type": "string",
        "description": "The number of consecutive health probe failures required
before moving the instance to the unhealthy state.",
        "pattern": "^[1-9]{1}[0-9]{0,2}$",
        "default": "10"
      },
      "LBCookieExpirationPeriod": {
        "type": "string",
        "description": "The time period, in seconds, after which the cookie should be
considered stale. If this parameter isn't specified, the sticky session will last for
the duration of the browser session.",
        "pattern": "^[0-9]+$|^$"
      },
      "LBCookieStickinessPolicyName": {
        "type": "string",
        "description": "A name for the load balancer cookie stickiness policy. The
name must be unique within the set of policies for this load balancer. To associate
with a listener, specify the name under PolicyNames in the respective listener
configuration.",
        "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
      },
      "AppCookieName": {
        "type": "string",
        "description": "A name for the application cookie used for stickiness.",
        "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
      },
      "AppCookiePolicyName": {
        "type": "string",
        "description": "A name for the application cookie stickiness policy. The
name must be unique within the set of policies for this load balancer. To associate
with a listener, specify the name under PolicyNames in the respective listener
configuration.",
        "pattern": "^[a-zA-Z0-9]{1,1}[a-zA-Z0-9-]{0,127}$|^$"
      }
    },
    "metadata": {
      "ui:order": [
        "Name",
```

```
          "Public",
          "SecurityGroups",
          "SubnetIds",
          "CrossZone",
          "IdleTimeout",
          "AccessLogInterval",
          "ConnectionDrainingTimeout",
          "HealthCheckHealthyThreshold",
          "HealthCheckInterval",
          "HealthCheckTarget",
          "HealthCheckTimeout",
          "HealthCheckUnhealthyThreshold",
          "LBCookieExpirationPeriod",
          "LBCookieStickinessPolicyName",
          "AppCookieName",
          "AppCookiePolicyName"
        ]
      },
      "required": [
        "SecurityGroups",
        "SubnetIds"
      ],
      "additionalProperties": false
    },
    "Listener1": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
```

```
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$",
          "default": "80"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ],
          "default": "HTTP"
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "SSLCertificateId"
        ]
      },
      "required": [
        "Port",
        "Protocol",
        "InstancePort"
      ],
      "additionalProperties": false
```

```
    },
    "Listener2": {
      "type": "object",
      "properties": {
        "InstancePort": {
          "type": "string",
          "description": "The TCP port the listener uses to send traffic to the target
 instance.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$|^$"
        },
        "InstanceProtocol": {
          "type": "string",
          "description": "The protocol the listener uses for routing traffic to back-
end connections (load balancer to backend instance).",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "Port": {
          "type": "string",
          "description": "The port number for the load balancer to use when routing
 external incoming traffic to the listener.",
          "pattern": "(?!^22$)(?!^3389$)(?!^5985$)^([1-9]{1}[0-9]{0,4})$|^$"
        },
        "Protocol": {
          "type": "string",
          "description": "The transport protocol to use for routing front-end
 connections (client to load balancer) to the listener.",
          "enum": [
            "HTTP",
            "HTTPS",
            "SSL",
            "TCP"
          ]
        },
        "SSLCertificateId": {
          "type": "string",
          "description": "The Amazon Resource Name (ARN) of the SSL
 certificate to use with the listener, in the form arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012.",
```

```
          "pattern": "^$|^arn:aws:acm:[a-z0-9-]+:[0-9]{12}:certificate/[0-9a-f]{8}-
[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$|^arn:aws:iam::[0-9]{12}:server-
certificate/.*$"
        }
      },
      "metadata": {
        "ui:order": [
          "Port",
          "Protocol",
          "InstancePort",
          "InstanceProtocol",
          "SSLCertificateId"
        ]
      },
      "additionalProperties": false
    }
  },
  "metadata": {
    "ui:order": [
      "Description",
      "VpcId",
      "Name",
      "TimeoutInMinutes",
      "StackTemplateId",
      "Tags",
      "AutoScaling",
      "LoadBalancer",
      "Listener1",
      "Listener2"
    ]
  },
  "required": [
    "Description",
    "VpcId",
    "Name",
    "TimeoutInMinutes",
    "StackTemplateId",
    "AutoScaling",
    "LoadBalancer",
    "Listener1"
  ],
  "additionalProperties": false
}
```

Document history

The following table describes the documentation for this release of AMS.

- **API version: 2019-05-21**
- **Latest documentation update:** May 23, 2024

Change	Description	Link(s)
New CT	Management | Advanced stack components | S3 storage | Add event notification. New CT.	ct-0o4zi9bzg74lp
Updated CT	Deployment | Standalone resources | EC2 instance | Create for WIGS (review required). Updated CT.	ct-36emj2uapfbu8
Deprecated CT	Deployment | Advanced stack components | AWS Identity and Access Management (IAM) | Create Entity or Policy (Read Permissions). Deprecated CT	
New CT	Management | Managed account | Automated IAM provisioning with read-write permissions | Update custom deny list (review required). New CT.	ct-2r9xvd3sdsic0
Updated CT	Management | Managed landing zone | Management account | Offboard application account. Updated CT.	ct-0vdiy51oyrhhm
New CT	Management | Managed Account | DNS | Migrate to Route 53.	ct-2tqi3kjcusen4

Change	Description	Link(s)
	New CT.	
Updated CT	Management | Advanced Stack Components | EBS Volume | Delete. Updated CT.	ct-3e3h8u0sp5z80
Updated CT	Deployment | Managed landing zone | Networking account | Add static route. Updated Additional Information section.	ct-3r2ckznmt0a59
Updated CT	Deployment | Advanced Stack Components | Identity and Access Management (IAM) | Create Service-Linked role. Added `CustomSuffix` input parameter.	ct-2eof6j3mlcwhf
Updated CT	Deployment | Managed Landing Zone | Management Account | Create Custom SCP (review required). Added a note in the Tips section.	ct-33ste5yc7hprs
New CT	Management | Advanced Stack Components | Route 53 Resolver | Disassociate resolver rules from VPC. New CT.	ct-2pfarpvczsstr
Updated CT	Management | Patching | Patch window | Update. Updated CT.	ct-2utx36abv83pv
Updated CT	Management | Advanced Stack Components | RDS database stack | Update deletion protection. Updated CT.	ct-2syhk4sr7cvyw
New CT	Management | Advanced Stack Components | RDS database stack | Update enhanced monitoring. New CT.	ct-3jx80fquylzhf

Change	Description	Link(s)
Updated CT	Deployment | Directory Service | DNS | Create Group Managed Service Account. Updated CT.	ct-2qhl8j1pjnbgn
Updated CT	Management | AWS Backup | Backup plan | Update. Updated CT.	ct-1ay83wy4vxa3k
Updated CT	Management | Advanced Stack Components | RDS database stack | Update master password. Updated CT.	ct-2052miu12d8fn
Updated CT	Deployment | Advanced Stack Components | ACM | Create private certificate. Updated CT.	ct-3ll9hnadql9s1
Updated CT	Deployment | Advanced Stack Components | EC2 stack | Create (with additional volumes). Updated Tips section.	ct-1aqsjf86w6vxg
Updated CT	Deployment | Advanced Stack Components | Database Migration Service (DMS) | Create Source Endpoint. Updated Tips section.	Database Migration Service (DMS) | Create Source Endpoint
New CT	Management | Advanced Stack Components | Route 53 Resolver | Associate VPC with Resolver Rule. New CT	ct-2pbqoffhclpek

Change	Description	Link(s)
Updated CT	Management | Advanced Stack Components | KMS key | Update. New CT	ct-3ovo7px2vsa6n
Updated CT	Management | Advanced Stack Components | Security group | Associate. New CT	ct-12lyw7otiyr6f
Updated CT	Management | Advanced Stack Components | Security group | Disassociate. New CT	ct-13lk0noacn6ua
Updated CT	Management | Advanced Stack Components | S3 storage | Update policy (review required). New CT	ct-0fpjlxa808sh2
Updated CT	Management | Advanced Stack Components | RDS database stack | Rotate DB certificate. New CT	ct-1ezarc5xph3tq
Updated CT	Management | Advanced Stack Components | Tag | Update (review required). New CT	ct-0zko7t3rk2efb
New CT	Management | Advanced stack components | KMS key | Share (review required). New CT	ct-05yb337abq3x5
Updated CT	Management | Advanced stack components | KMS key | Update (review required). New version	ct-3ovo7px2vsa6n

Change	Description	Link(s)
New CT	Management | Directory Service | Directory | Create AD trust.	ct-0x6dylrnfjgz5
Updated CT	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create access key. Version 2.	ct-2hhqzgxvkcig8
Updated CT	Deployment | Advanced stack components | Redshift | Create (cluster subnet group). New required parameterSubnetGroupDescription.	ct-0q43l40hxrzum
Updated CTs	Deployment | Advanced stack components | EC2 Stack | Create. Deployment | Ingestion | Stack from migration partner migrated instance | Create. New optional parameter EnforceIMDSv2.	ct-14027q0sjyt1h ct-257p9zjk14ija
New CT	Management | Managed account | Stack access duration | Override (review required).	ct-0jb01cofkhwk1
New CTs	Deployment | Advanced Stack Components | Identity and Access Management (IAM) | Create entity or policy (read-write permissions). Management | Advanced Stack Components | Identity and Access Management (IAM) | Update entity or policy (read-write permissions). Management | Advanced Stack Components | Identity and Access Management (IAM) | Delete entity or policy (read-write permissions). Management | Managed account | Automated IAM provisioning with read-write permissions | Enable (review required)	ct-1n9gfnog5x7fl ct-1e0xmuy1diafq ct-17cj84y7632o6 ct-1706xvvk6j9hf

Change	Description	Link(s)
New CTs	Deployment | Managed landing zone | Management account | Create stacksets stack, ct-16pknsfa8lul7. Management | Managed landing zone | Management account | Delete stacksets stack, ct-1yqy4frl5s8y8. Management | Managed landing zone | Management account | Update stacksets stack, ct-1v9g9n30woc8h.	ct-16pknsfa8lul7 ct-1yqy4frl5s8y8 ct-1v9g9n30woc8h
Updated CT	Management | Access | Stack admin access | Grant, ct-1dmlg9g1l91h6. Version 3.0 adds support for a custom maximum stack access time. Submit an RFC with the Management | Other | Other | Update (review required) (ct-0xdawir96cy7k) change type in the SALZ account or MALZ Shared Services account to customize this value.	ct-1dmlg9g1l91h6
Updated CT	Management | Access | Stack admin access | Update, ct-0ikpop8zqhkxg. Version 3.0 adds support for a custom maximum stack access time. Submit an RFC with the Management | Other | Other | Update (review required) (ct-0xdawir96cy7k) change type in the SALZ account or MALZ Shared Services account to customize this value.	ct-0ikpop8zqhkxg
Updated CT	Management | Access | Stack read-only access | Grant, ct-199h35t7uz6jl. Version 3.0 adds support for a custom maximum stack access time. Submit an RFC with the Management | Other | Other | Update (review required) (ct-0xdawir96cy7k) change type in the SALZ account or MALZ Shared Services account to customize this value.	ct-199h35t7uz6jl

Change	Description	Link(s)
Updated CT	Management | Access | Stack read-only access | Update, ct-3kh1wiizlne1i. Version 3.0 adds support for a custom maximum stack access time. Submit an RFC with the Management | Other | Other | Update (review required) (ct-0xdawir96cy7k) change type in the SALZ account or MALZ Shared Services account to customize this value.	ct-3kh1wiizlne1i
Updated CT	Management | Access | Stack admin access | Grant, ct-1dmlg9g1l91h6. Version 3.0 is the default version, adding support for multiple usernames and access times up to 12 hours (previously 8).	ct-1dmlg9g1l91h6
Updated CT	Management | Access | Stack admin access | Update, ct-0ikpop8zqhkxg. Version 3.0 is the default version, adding support for multiple usernames and access times up to 12 hours (previously 8).	ct-0ikpop8zqhkxg
Updated CT	Management | Access | Stack read-only access | Grant, ct-199h35t7uz6jl. Version 3.0 is the default version, adding support for multiple usernames and access times up to 12 hours (previously 8).	ct-199h35t7uz6jl
Updated CT	Management | Access | Stack read-only access | Update, ct-3kh1wiizlne1i. Version 3.0 is the default version, adding support for multiple usernames and access times up to 12 hours (previously 8).	ct-3kh1wiizlne1i
New CT	Management | Advanced stack components | Bastions | Update bastion instance size (review required), ct-0tmpmp1wpgkr9.	ct-0tmpmp1wpgkr9

Change	Description	Link(s)
New CT	Management | Advanced stack components | Bastions | Update bastion instance or session counts (review required) ct-1962s5oczal9z.	ct-1962s5oczal9z
New CT	Management | Advanced stack components | Bastions | Add CIDR Ingress (review required) ct-36zubwzxp44a4.	ct-36zubwzxp44a4
New CT	Management | Advanced stack components | EC2 instance stack | Associate private ip addresses (review required), ct-1pvlhug439gl2.	ct-1pvlhug439gl2
New CT	Management | Advanced stack components | EC2 instance stack | Update instance detailed monitoring, ct-0tmpmp1wpgkr9.	ct-0tmpmp1wpgkr9
New CT	Deployment | Advanced stack components | VPC | Add static route (review required), ct-06bwg93ukgg8t.	ct-06bwg93ukgg8t
Updated CT	Management | Advanced stack components | AMI | Deregister (multiple), ct-26vhhlj9jmlpf Version 2.0 supports deregistering multiple AMIs in one request.	ct-26vhhlj9jmlpf
Updated CT	Management | Advanced stack components | Security group | Authorize ingress rule, ct-3j2zstluz6dxq. Updated to version 3.0.	ct-3j2zstluz6dxq
Updated CT	Management | Advanced stack components | Security group | Revoke ingress rule, ct-1vjbacfr4ufdv. Updated to version 3.0.	ct-1vjbacfr4ufdv
Updated CT	Deployment | AMS Resource Scheduler | Solution | Deploy, ct-0ywnhc8e5k9z5. Version 2 requires the Action (deploy or update) parameter.	ct-0ywnhc8e5k9z5

Change	Description	Link(s)
Updated CT	Management | AMS Resource Scheduler | Solution | Update, ct-2c7ve50jost1v. Version 2 requires the Action (deploy or update) parameter.	ct-2c7ve50jost1v
Updated CT	Deployment | Advanced stack components | Auto Scaling group | Create, ct-2tylseo8rxfsc. New optional parameter EnforceIMDSv2.	ct-2tylseo8rxfsc
Updated CT	Deployment | Standard stacks | High availability two-tier stack | Create, ct-06mjngx5flwto. New optional parameter EnforceIMDSv2.	ct-06mjngx5flwto
New CT	Management | Directory Service | Directory | Share Directory, ct-369odosk0pd9w.	ct-369odosk0pd9w
New CT	Management | Directory Service | Directory | Unshare Directory , ct-2xd2anlb5hbzo.	ct-2xd2anlb5hbzo
Updated CT	Management | Host security | Trend Micro DSM | Add login (read-only) , ct-0wspy4o646g9p. Version 2 has new parameters and examples. It does not require a review.	ct-0wspy4o646g9p
Updated CT	Management | Advanced stack components | AMI | Deregister, ct-26vhhlj9jmlpf. Version 2 supports deleting or deregistering multiple AMIs.	ct-26vhhlj9jmlpf
Updated CT	Deployment | Directory Service | DNS | Create conditional forwarder, ct-3nba0wtdugnan. Conditional forwarder now supports up to 5 IP addresses.	ct-3nba0wtdugnan

Change	Description	Link(s)
Updated CT	Management | Directory Service | DNS | Update conditional forwarder, ct-2fqmbyud166z9. Conditional forwarder now supports up to 5 IP addresses.	ct-2fqmbyud166z9
Updated CT	Management | Advanced stack components | EC2 instance stack | Change hostname (Linux), ct-2781aqd6f6svs. Version 2 does not require a review.	ct-2781aqd6f6svs
Updated CT	Deployment | Advanced stack components | Auto Scaling group | Create, ct-2tylseo8rxfsc. Version 3 has new parameter descriptions.	ct-2tylseo8rxfsc
Updated CT walkthrough	Management | Managed landing zone | Application account | Confirm offboarding, ct-2wlfo2jxj2rkj. Added tips.	ct-2wlfo2jxj2rkj
Updated CT walkthroughs	Management | Advanced stack components | DNS (private) | Update, ct-1d55pi44ff21u and Deployment | Advanced stack components | DNS (private) | Create, ct-0c38gftq56zj6. Added tips about 500 limit on RRs.	ct-1d55pi44ff21u and ct-0c38gftq56zj6
New CT	Deployment | Advanced stack components | VPCEndpoint (Interface) | Create, ct-3oafsdbzjtuqp.	ct-3oafsdbzjtuqp
New CT	Deployment | Advanced stack components | S3 storage | Update encryption, ct-128svy9nn2yj8.	ct-128svy9nn2yj8
New CT	Deployment | Advanced stack components | S3 storage | Update versioning, ct-2hh93eyzmwbkd.	ct-2hh93eyzmwbkd
New CT	Deployment | Advanced stack components | RDS database stack | Update instance type, ct-13swbwdxg106z.	ct-13swbwdxg106z

Change	Description	Link(s)
New CT	Deployment | Advanced stack components | RDS database stack | Update MultiAZ setting, ct-36jq7gvwyty8h.	ct-36jq7gvwyty8h
New CT	Deployment | Advanced stack components | RDS database stack | Update storage, ct-0loed9dzig1ze.	ct-0loed9dzig1ze
Updated CT	Deployment | Advanced stack components | RDS snapshot | Copy, ct-1c0jrxd3su5oe. Added support for multi-region (MRK) KMS keys.	ct-1c0jrxd3su5oe
Updated CT	Management | Advanced stack components | EC2 instance stack | Replace instance profile, ct-37kcp2v1mriu6. Updated to version 2.0 with new parameters.	ct-37kcp2v1mriu6
Updated CT	Deployment | Advanced stack components | EC2 stack | Create (with additional volumes), ct-1aqsjf86w6vxg. Added support for specifying core count and threads per core.	ct-1aqsjf86w6vxg
Updated CT	Deployment | Advanced stack components | KMS key | Create, ct-1d84keiri1jhg. Added guidance for deleting a key, and a warning that deletion does not occur automatically.	ct-1d84keiri1jhg
Updated CTs	Deployment | Advanced stack components | Tag | Create, ct-3cx7we852p3af. Updated parameters, description, and console screenshot.	ct-3cx7we852p3af
Updated CTs	Management | Advanced stack components | Tag | Update, ct-0xqwmtn1hfh8u. Updated parameters, description, and console screenshot.	ct-0xqwmtn1hfh8u

Change	Description	Link(s)
Updated CTs	Management | Advanced stack components | Tag | Delete, ct-2zebb2czoxpjd. Updated parameters, description, and console screenshot.	ct-2zebb2czoxpjd
Updated CTs	Management | Standalone resources | EC2 instance | Terminate , ct-3dfubbpesm2v9. Updated parameters, description, and console screenshot.	ct-3dfubbpesm2v9
Updated CT	Management | Advanced stack components | EC2 instance stack | Update DeleteOnTermination (review required), ct-2aaaqid7asjy6. Updated description of the DeleteOnTermination parameter.	ct-2aaaqid7asjy6
Updated CT	Management | Advanced stack components | KMS key | Update, ct-3ovo7px2vsa6n. Updated the description of the KeyRotation parameter.	ct-3ovo7px2vsa6n
Updated CT	Management | Advanced stack components | RDS database stack | Update master user password, ct-2052miu12d8fn. Updated examples and tips with additional guidance for using SSM Parameter Store or AWS Secrets Manager	ct-2052miu12d8fn
Updated CT	Management | Advanced stack components | Target group | Detach instances, ct-37bq2l9c8fzxv. Updated description of InstancesIds parameter.	ct-37bq2l9c8fzxv

Change	Description	Link(s)
Updated CTs	Added support for GP3 storage, and choice of SQL character sets to: • Deployment | Advanced stack components | RDS database stack | Create • Management | Advanced stack components | RDS database stack | Update	ct-2z60dyvto9g6c and ct-12w49boaiwtzp
Updated CTs	Added support for multi-Region KMS keys (MRK) to: • Deployment | Advanced stack components | S3 storage | Create • Management | Advanced stack components | S3 storage | Update	ct-1a68ck03fn98r and ct-1gi93jhvj28eg
CT Name Change	Deployment | Advanced stack components | OpenSearch | Create domain, ct-0azen3a9anxzj is removed and replaced with Deployment | Advanced stack components | OpenSearch | Create domain, ct-281et7bs9ep4s. This update included a change to the OpenSearch | Create domain schema.	ct-281et7bs9ep4s
New CT	Management | Managed landing zone | Networking account | Remove TGW static route, ct-0rmgrnr9w8mzh.	ct-0rmgrnr9w8mzh
New CT	Deployment | AMS Resource Scheduler | Solution | Deploy, ct-2c7ve50jost1v.	ct-0ywnhc8e5k9z5

Change	Description	Link(s)
Updated CTs	Management | Advanced stack components | DNS (private) | Create, ct-0c38gftq56zj6. Management | Advanced stack components | DNS (private) | Update, ct-1d55pi44ff21u. Management | Advanced stack components | DNS (public) | Create, ct-0vzsr2nyraedl. Management | Advanced stack components | DNS (public) | Update, ct-1hzofpphabs3i. Parameters and CLI examples have changed for version 2.0.	ct-0c38gftq56zj6, ct-1d55pi44ff21u, ct-0vzsr2nyraedl, and ct-1hzofpphabs3i
Updated CTs	Deployment | Advanced stack components | RDS database stack | Create (for Aurora), ct-2jvzjwunghrhy. Deployment | Advanced stack components | RDS database stack | Create from backup (for Aurora), ct-2wllq61djysxz. Management | Advanced stack components | RDS database stack | Update (for Aurora), ct-2dphvdy1krpj6. Updated descriptions, added min/max parameters serverless scaling, and added InstanceType values.	ct-2jvzjwunghrhy, ct-2wllq61djysxz, and ct-2dphvdy1krpj6
Updated CT	Management | Advanced stack components | EC2 instance stack | Resize, ct-15mazjj88xc69. Updated description, parameters, and CLI examples for version 2.0.	ct-15mazjj88xc69
New CT	Management | AMS Resource Scheduler | Solution | Update, ct-2c7ve50jost1v.	ct-2c7ve50jost1v

Change	Description	Link(s)
Updated CT	Management| AMS Resource Scheduler | Solution | Deploy, ct-0ywnhc8e5k9z5. Removed SALZ restriction and added Action parameter.	ct-0ywnhc8e5k9z5
New CT	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create access key, ct-2hhqzgxvkcig8.	ct-2hhqzgxvkcig8
New CT	Management | Advanced stack components | Identity and Access Management (IAM) | Deleate or deactivate access key, ct-37qquo9wbpa8x.	ct-37qquo9wbpa8x
Updated CT	Deployment | Advanced stack components | Application Load Balancer | Create, ct-111r1yayblnw4. Added parameter TargetGroup to optionally specify a load balancer TargetGroup.	ct-111r1yayblnw4
Updated CT	Management | Managed landing zone | Application account | Confirm Offboarding, ct-2wlfo2jxj2rkj. Added a tip not to use this for Customer Managed application accounts.	ct-2wlfo2jxj2rkj
Updated CT	Management | Managed landing zone | Management account | Offboard application account, ct-0vdiy51oyrhhm. Added a tip: when applied to Customer Managed application accounts, there is no confirmation step.	ct-0vdiy51oyrhhm
Updated CT	Deployment | Advanced stack components | KMS alias | Create, ct-2svg4k2fqi4ak. Updated the AliasName pattern to reject names with prohibited prefixes.	ct-2svg4k2fqi4ak

Change	Description	Link(s)
New Content	The What Are AMS Change Types? file now has a link to a zip file with a current comma-separated value (CSV) file of change types.	Change type CSV output file
New CT	Management | Advanced stack components | EBS snapshot | Archive, ct-059ewa92tc2i1.	ct-059ewa92tc2i1
New CT	Management | Advanced stack components | Identity and Access Management (IAM) | Update max session duration, ct-1fzddqrr20c2i.	ct-1fzddqrr20c2i
New CT	Deployment | Advanced stack components | RDS snapshot | Create (Cluster), ct-2zqwr34epwzx1.	ct-2zqwr34epwzx1
New CT	Management | Standalone resources | RDS instance | Terminate , ct-3glr80c15rp7z.	ct-3glr80c15rp7z
New CT	Management | Advanced stack components | RDS snapshot | Delete, ct-0idxb0xsg1ui6.	ct-0idxb0xsg1ui6
New CT	Deployment | Managed landing zone | Networking account | Create transit gateway route table, ct-3dscwaeyi6cup.	ct-3dscwaeyi6cup
New CT	Management | Advanced stack components | S3 storage | Manage lifecycle configuration, ct-1ax768xtu8c9q.	ct-1ax768xtu8c9q
Updated CT	Deployment | Advanced stack components | EC2 stack | Create, ct-14027q0sjyt1h. Added a tip not to select instance types that are too small.	ct-14027q0sjyt1h
Updated CT	Management | Advanced stack components | Application Load Balancer | Update, ct-1a1zzgi2nb83d. Updated the description of the LoadBalancerSubnetIds parameter.	ct-1a1zzgi2nb83d

Change	Description	Link(s)
Updated CT	Deployment | Patching | SSM patch window | Create, ct-0el2j07llrxs7. Replaced references to Route 53 hosted zones with SSM window creation.	ct-0el2j07llrxs7
Updated CT	Management | AWS service | Self-provisioned service | Add, ct-1w8z66n899dct. List of service names updated to include AWS Private Certificate Authority (PCA)	ct-1w8z66n899dct
Updated CT	Management | AWS service | Self-provisioned service | Add (review required), ct-3qe6io8t6jtny. List of service names updated to include AWS Private Certificate Authority (PCA)	ct-3qe6io8t6jtny
Updated CT	Management | Patching | On demand patching | Run, ct-3oy53m1qzl2s5. Added a note regarding the StartInactiveInstances parameter. Inactive instances return to their inactive state after patching.	ct-3oy53m1qzl2s5
Updated CT	Management | Advanced stack components | Tag | Bulk update, ct-3047c34zuvswh. Added Supported Resources section and a link to Tag bulk update notes.	ct-3047c34zuvswh
Updated CT	Management | Advanced stack components | Tag | Bulk update (review required), ct-0k4b96aatyqgl. Added Supported Resources section and a link to Tag bulk update notes.	ct-0k4b96aatyqgl
New CT	Deployment | Advanced stack components | RDS snapshot | Copy (for Aurora), ct-19fdy7np55xiu.	ct-19fdy7np55xiu

Change	Description	Link(s)
New CT	Management | Advanced stack components | RDS database stack | Start Aurora Cluster, ct-02ocqy2i0jx3t.	ct-02ocqy2i0jx3t
New CT	Management | Advanced stack components | RDS database stack | Stop Aurora Cluster, ct-37vqa0oggka3q.	ct-37vqa0oggka3q
Updated CT schema	Management | AWS service | Self-provisioned service | Add, ct-1w8z66n899dct. The AWS Codepipeline service was removed from the possible services you could add with this CT.	ct-1w8z66n899dct
Updated CT schema	Deployment | Advanced stack components | AMI | Copy, ct-046aizcwg5idf. The **Region** parameter was updated to add: "This must be the account onboarded Region."	ct-046aizcwg5idf
Updated CT schemas	Management | Directory Service | DNS | Add A record, ct-2w3rbmnny1qpo Management | Directory Service | DNS | Add CNAME record, ct-2murl5xzbxoxf Management | Directory Service | DNS | Remove record, ct-1icrtx8ydvdwe Improved examples.	ct-2w3rbmnny1qpo ct-2murl5xzbxoxf ct-1icrtx8ydvdwe
Updated CT tips	Deployment | Advanced stack components | RDS database stack | Create from snapshot, ct-20san5sgtwd9e. Added a note: "You can't restore a DB instance from a DB snapshot that is both shared and encrypted. Instead, you can make a copy of the DB snapshot and restore the DB instance from the copy. To copy the shared snapshot, please use the following CT: RDS Snapshot | Copy".	ct-20san5sgtwd9e

Change	Description	Link(s)
Updated CT schemas	Deployment/Advanced stack components/Tag/Create, ct-3cx7we852p3af. Management/Advanced stack components/Tag/Delete, ct-2zebb2czoxpjd. Management/Advanced stack components/Tag/Update, ct-0xqwmtn1hfh8u. For all three, the **ResourceArns** parameter description was updated.	ct-3cx7we852p3af ct-2zebb2czoxpjd ct-0xqwmtn1hfh8u
Updated CT tips	Management | Directory Service | DNS | Add A record, ct-2w3rbmnny1qpo. Management | Directory Service | DNS | Add CNAME record, ct-2murl5xzbxoxf. Added a note: "For multi-account landing zone (MALZ), use this change type in the shared services account."	ct-2w3rbmnny1qpo ct-2murl5xzbxoxf
Updated content	Updated **Tips** section with information on "Linux Preparation for AMI Create," "Windows Preparation for AMI Create," and "UserData for AMI Create."	AMI | Create.
New content	Example walkthroughs for each change type have been moved here from the retired *AMS Advanced Change Management Guide*. All change types with execution mode=manual are now appended with "(review required)" to the Operation name. All change types with execution mode=automated are now appended with nothing (any cases of "(auto)" or "(no review required)" are removed).	Change Types by Classification.
New CT	Management | Managed account | Direct Change mode | Enable, CT ID: ct-3rd4781c2nnhp.	ct-3rd4781c2nnhp

Change	Description	Link(s)
Updated CTs	Change types that included "Master account" in their classific ation have all been updated to "Management account."	Managed Landing Zone Subcategory
	Removed the space in pattern for parameter: `IpProtocol .`	ct-3j2zst luz6dxq ct-0lqrua jvhwsbk ct-111fhp lhx9axe ct-1vjbac fr4ufdv
New CTs:	Management | Advanced stack components | EC2 instance stack | Change time zone (ct-3g9dbtun44mal)	ct-3g9dbt un44mal
	Management | Advanced stack components | RDS database stack | Update deletion protection	ct-2syhk4 sr7cvyw
	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create service-specific credentials	ct-2ni31o yto1i5k
	Management | Advanced stack components | Identity and Access Management (IAM) | Reset service-specific credentials	ct-22cbvc 1yujhec
Updated CTs:	Deployment| Managed landing zone | Management Account | Create custom SCP (review required) (ct-33ste5yc7hprs) and Deployment| Managed landing zone | Networking account | Create application route table (review required) (ct-1urj9 4c3hdfu5) Change classification to include "(review required)" appended to the Operation name.	ct-33ste5 yc7hprs and ct-1urj94 c3hdfu5

Change	Description	Link(s)
	Deployment | Advanced stack components | Database Migration Service (DMS) | Start replication task (ct-1yq7hhqse71yg) Updated to indicate the **DocumentName** and **Region are required parameters**; previously, they were erroneously listed as optional.	ct-1yq7hhqse71yg
	Deployment | AWS Backup | Backup plan | Create (ct-2hyozbpa0sx0m) Added note to warn that not all resources types supported by AWS Backup are enabled by default. Find which services are in Getting Started 1: Service Opt-In. If changes need to be made, open an RFC using CT ct-1e1xtak34nx76.	ct-2hyozbpa0sx0m
	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create EC2 instance profile (ct-117rmp64d5mvb) and Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Lambda execution role (ct-1k3oui719dcju) New Version: 2.0. Updated to make JSON copy-paste easier.	ct-117rmp64d5mvb and ct-1k3oui719dcju
	Deployment | Advanced stack components | RDS database stack | Create (ct-2z60dyvto9g6c) Added a new value for the **RDSDBEngine** parameter: mariadb.	ct-2z60dyvto9g6c
	Deployment | Advanced stack components | RDS database stack | Update (ct-12w49boaiwtzp) Extended the expected duration time to 360 minutes from 60 minutes.	ct-12w49boaiwtzp

Change	Description	Link(s)
Updated CTs "Additional Information"	Added an important note about limitations on the resources that can be remediated.	ct-34sxfo53yuzah
New CTs:	Management | AWS service | Self-provisioned service | Add Self-Service Provisioning service (no review required)	ct-1w8z66n899dct
	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create service-linked role	ct-2eof6j3mlcwhf
	Management | Advanced stack components | ACM | Delete certificate	ct-1q8q56cmwqj9m
	Management | Managed landing zone | Application account | Confirm offboarding	ct-2wlfo2jxj2rkj
	Management | Managed landing zone | Application account | Offboard application account	ct-0vdiy51oyrhhm
	Deployment | Managed landing zone | Management account | Create Accelerate account	ct-0vdiy51oyrhhm
Updated CTs:		

Change	Description	Link(s)
Management | AMS Resource Scheduler | Schedule | Add (ct-2bxelbn765ive) and Management | AMS Resource Scheduler | Schedule | Update (ct-3u61cd4edns0x) The **SSMMaintenanceWindow** can now take a list of AWS Systems Manager existing maintenence windows.	ct-2bxelbn765ive and ct-3u61cd4edns0x	
New CTs:	Deployment | Advanced stack components | Identity and Access Managment (IAM) | Create OpenID Connect provider	ct-30ecvfi3tq4k3

Change	Description	Link(s)
Updated CTs:	Deployment | Advanced stack components | RDS database stack | Create This update adds performance insights options.	ct-2z60dyvto9g6c
Updated CTs "Additional Information"	Many CTs had non-standard "Additional Information" sections without links to the corresponding walkthrough, this has been fixed.	January 27, 2022
New CTs:	Management | Advanced stack components | Application Load Balancer | Add listener certificate ct-3g6fq83nxg1a7	January 13, 2022
	Management | Advanced stack components | Application Load Balancer | Remove listener certificate ct-0tpbr6lfa3zng	
	Management | Advanced stack components | Load Balancer (ELB) stack | Replace listener certificate ct-0aqx5t0pgfzbg	
	Management | Advanced stack components | Network Load Balancer | Add listener certificate ct-35p977vul06df	
	Management | Advanced stack components | Network Load Balancer | Remove listener certificate ct-3929xwf222jri	
	Management | Managed landing zone | Networking account | Disable TGW propagation ct-2pxyajek47am2	

Change	Description	Link(s)
	Management | Managed landing zone | Networking account | Enable TGW propagation ct-1f9hi4bephqa9	
	Management | Standalone resources | EC2 instance | Terminate ct-3dfubbpesm2v9	
Updated CTs:	Management | Advanced stack components | EC2 Instance Stack | Gather log4j information (v2.0) ct-19f40lfm5umy8	This update provides an option to target all instances in the region.
	Management | Advanced stack components | Database Migration Service (DMS) | Start replication task and Management | Advanced stack components | Database Migration Service (DMS) | Stop replication task ct-1yq7hhqse71yg and ct-1vd3y4ygbqmfk	This change updates the task ARN regular expression to the allow tasks containing the a dash (-).
	Management | Advanced stack components | EC2 instance stack | Change hostname (Linux) ct-2781aqd6f6svs	Automated, with additional parameters, and moved to version 2.0.

Change	Description	Link(s)
	Management | Advanced stack components | EC2 instance stack | Restore volumes ct-0ffvihqwjvqj1	This update adds two optional parameters, **RootVolumeType** and **VolumeTypes**.

Change	Description	Link(s)
	A **Priority** parameter has been added to all `execution mode=manual` change types (CTs). For additional information, see RFC scheduling.	The changed CTs are: `Deployment/Monitoring and notification/GuardDuty IP set/CreateDeployment/Monitoring and notification/GuardDuty threat intel set/CreateManagement/Monitoring and notification/GuardDuty IP set/DeleteManagement/Monitoring`

Change	Description	Link(s)
		and notification/GuardDuty IP set/UpdateManagement/Monitoring and notification/GuardDuty threat intel set/DeleteManagement/Monitoring and notification/GuardDuty threat intel set/UpdateManagement/Other/Other/CreateManagement/Other/

Change	Description	Link(s)
		Other/Updated Deployment/Managed landing zone/Management account/Create Custom SCP Deployment/Managed landing zone/Networking account/Create application route table Deployment/Advanced stack components/Identity and Access Management (IAM)/Create entity or policy Deployment/

Change	Description	Link(s)
		Advanced stack components/KMS key/Create (review required) (1.0 and 2.0) Deployment/Advanced stack components/S3 storage/Create policy Deployment/Advanced stack components/Security group/Create (review required) Deployment/Advanced stack components/Tag/Create (review required) Management/AWS service/

Change	Description	Link(s)
		Self-provisioned service/Add Management/Advanced stack components/EC2 instance stack/Change hostname (Linux) (1.0) Management/Advanced stack components/Identity and Access Management (IAM)/Delete entity or policy Management/Advanced stack components/Identity and Access Managemen

Change	Description	Link(s)
		t (IAM)/ Update entity or policy Ma nagement/ Advanced stack component s/KMS key/ Delet e(1.0 and 2.0) Mana gement/ Advanced stack component s/KMS key/ Updat e(1.0 and 2.0) Mana gement/ Advanced stack component s/S3 storage/ Delete policy Ma nagement/ Advanced stack component s/S3 storage/

Change	Description	Link(s)
		Update policy Management/Advanced stack components/Security group/Delete Management/Advanced stack components/Security group/Update Management/Advanced stack components/Tag/Bulk update (review required) Management/Advanced stack components/Tag/Delete (review required) Management/Advanc

Change	Description	Link(s)
		ed stack component s/Tag/ Update (review required) Manageme nt/ Managed account/ D eveloper mode/ Enable Manage ment/ Standard stacks/ Stack/ Remed iate drift Man agement/ A pplicatio ns/IAM instance profile/ Create Man agement/ Host security/ Malware full system scan/ Disable

Change	Description	Link(s)
New CTs:	Management | Advanced Stack Components | EC2 Instance Stack | Gather log4j information Single instance scan use v1.0, Multi instance scan use v2.0. ct-19f40lfm5umy8	December 20, 2021
New CTs:	Management | Directory service | DNS | Delete conditional forwarder ct-1icghmq38rnsn	December 17, 2021
	Management | Directory service | DNS | Update conditional forwarder ct-2fqmbyud166z9	December 17, 2021
New CTs:	Deployment | Directory service | DNS | Create conditional forwarder ct-3nba0wtdugnan	November 30, 2021
	Deployment | Directory service | DNS | Create group managed service account ct-2qhl8j1pjnbgn	November 30, 2021
	Management | Directory service | DNS | Update record permission ct-1eft8s6vdhz0w	November 30, 2021
	Management | Directory service | DNS | Update cluster permissions ct-03ytgoevfebjr	November 30, 2021

Change	Description	Link(s)
	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create EC2 instance profile ct-117rmp64d5mvb	November 30, 2021
	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create Lambda execution role ct-1k3oui719dcju	November 30, 2021
New CTs:	Management | Managed landing zone | Management account | Move Account to OU ct-1vq0f289r36ay	October 28, 2021
Updated CTs:	Deployment | AWS Backup | Backup plan | Create ct-2hyozbpa0sx0m. Additional parameters were for creating backup copies across accounts.	November 11, 2021
	Management | Advanced stack components | EBS snapshot | Delete ct-30bfiwxjku1nu. The CT description was expanded to mention some cavaets.	October 28, 2021
	Management | Advanced stack components | EC2 instance | Start ct-03t7kvuwx6rgr. The CT schema was updated so you can start mulitple EC2 instances.	
	Management | Advanced stack components | EC2 instance | Stop ct-3mvvt2zkyveqj. The CT schema was updated so you can stop mulitple EC2 instances. There's also a new parameter, **ForceStop**.	

Change	Description	Link(s)
	Management | AWS Backup | Recovery point | Delete ct-1r1vbr8ahr156. The CT schema was updated so you can delete mulitple recovery points.	
Updated CTs:	Deployment | Advanced stack components | Redshift | Create (cluster from snapshot) version 1.0 ct-3jrqmeq7j0wke. A new parameter, NodeType, was added.	October 14, 2021
Missing Examples	Many change type examples were missing, this has been fixed.	September 30, 2021
New CTs:	Management | Directory Service | Computer object | Remove SPN ct-1078jhyxq32dp	September 30, 2021
	Deployment | Managed landing zone | Networking account | Disassociate TGW attachment ct-3jo8yccbin4it	September 30, 2021
	Deployment | Managed landing zone | Networking account | Associate TGW attachment ct-3nmhh0qr338q6	September 30, 2021
	Deployment | Managed landing zone | Networking account | Add static route ct-3r2ckznmt0a59	September 30, 2021
	Deployment | Managed landing zone | Management account | Create Developer Mode account (with VPC) ct-38xcr0q86k9lh	September 30, 2021

Change	Description	Link(s)
	Management | Advanced stack components | Security Group | Delete (no review required) ct-18r16ldqil6w9	September 30, 2021
	Management | Advanced stack components | Security Group | Disassociate (no review required) ct-13lk0noacn6ua	September 30, 2021
Updated CTs:	Management | Advanced stack components | AMI | Deregister ct-26vhhlj9jmlpf This CT has a new parameter "DeleteSnapshots" to allow deleting snapshots associated with AMI.	September 30, 2021
	Management | Advanced stack components | AMI | Deregister ct-2r2bffv9u6q4m A note was added that you cannot use the CT with Aurora MySQL or Aurora PostgreSQL.	September 30, 2021
New CTs:	Classification (two): Management | Standard stacks | Stack | Remediate drift (auto) Management | Custom stack | Stack | Remediate drift (auto) ct-3kinq0u4l33zf.	September 16, 2021
	Management | AWS Backup | Backup plan | Enable cross account copy (Management account) ct-2yja7ihh30ply.	September 16, 2021
	Management | Advanced stack components | EC2 instance stack | Encrypt instance volumes ct-0hahohe17csnc.	September 16, 2021
Updated CTs:	Deployment | Advanced stack components | Database Migration Service (DMS) | Create replication subnet group This CT (ct-2q5azjd8p1ag5) will fail if the 'dms-vpc-role' IAM role doesn't exist in the account. ct-2q5azjd8p1ag5.	September 16, 2021

Change	Description	Link(s)
	Deployment | Advanced stack components | EC2 stack | Create The note for **InstanceType** has been updated to this "Any EC2 instance created within AMS environments have pre-configured AMS components and agents like EPS, SSM, Cloudwatch etc., which occupy the resource's capacity along with the applicati on workload. Therefore AMS does not recommend using the t2.micro/t3.micro and t2.nano/t3.nano instance types as they can impact the performance of the application and AMS tooling running on the instances. For more information, see Choosing the Right EC2 Instance Type for Your Application. ct-14027q0sjyt1h.	September 16, 2021
	Deployment | Advanced stack components | EC2 stack | Create (with additional volumes) The note for **InstanceType** has been updated to this "Any EC2 instance created within AMS environments have pre-configured AMS components and agents like EPS, SSM, Cloudwatch etc., which occupy the resource's capacity along with the applicati on workload. Therefore AMS does not recommend using the t2.micro/t3.micro and t2.nano/t3.nano instance types as they can impact the performance of the application and AMS tooling running on the instances. For more information, see Choosing the Right EC2 Instance Type for Your Application. ct-1aqsjf86w6vxg.	September 16, 2021
	Management | Advanced stack components | Security Group | Associate There is a new version and support for additional resource types. ct-12lyw7otiyr6f.	September 16, 2021

Change	Description	Link(s)
New information, a note about the difference in usage for the change types if using a single-account landing zone or multi-account landing zone account.	Deployment | AMS Resource Scheduler | Solution | Deploy ct-0ywnhc8e5k9z5. Management | AMS Resource Scheduler | State | Enable ct-2wrvu4kca9xky. Management | AMS Resource Scheduler | State | Disable ct-14v49adibs4db.	August 26, 2021
New CTs:	Management | Patching | Patch window | Update ct-2utx36abv83pv.	August 26, 2021
	Management | Advanced stack components | KMS key | Enable rotation ct-2lt0jeydeumpe.	August 26, 2021
	Management | Directory Service | Users and groups | Add group to group ct-1i20abktsm05v.	August 26, 2021
	Management | Directory Service | Users and groups | Add user to group ct-24pi85mjtza8k.	August 26, 2021
	Management | Directory Service | Users and groups | Add group ct-3eutt7grkict4.	August 26, 2021
	Management | Directory Service | Users and groups | Remove user from group ct-2019s9y3nfml4.	August 26, 2021
	Management | Directory Service | DNS | Remove record ct-1icrtx8ydvdwe.	August 26, 2021

Change	Description	Link(s)
	Management | Directory Service | DNS | Add CNAME record ct-2murl5xzbxoxf.	August 26, 2021
	Management | Directory Service | DNS | Add A record ct-2w3rbmnny1qpo.	August 26, 2021
Updated CTs:	Management | Advanced stack components | EC2 instance stack | Restore volumes The schema is updated with new parameters and the version is now 3.0. ct-2z60dyvto9g6c.	August 26, 2021
	Deployment | Advanced stack components | DNS (private) | Create The schema is updated with new parameters: **AliasTarg etDnsName**, **AliasTargetHostedZoneId**, and **AliasTarg etEvaluatedTargetHealth** to support "A" record to route traffic to AWS resource such as CloudFront distribution or an Amazon S3 bucket, by providing the DNSName and HostedZoneID associated with the AWS resource. ct-0c38gftq56zj6 .	August 26, 2021
	Deployment | Advanced stack components | DNS (public) | Create The schema is updated with new parameters: **AliasTarg etDnsName**, **AliasTargetHostedZoneId**, and **AliasTarg etEvaluatedTargetHealth** to support "A" record to route traffic to AWS resource such as CloudFront distribution or an Amazon S3 bucket, by providing the DNSName and HostedZoneID associated with the AWS resource. ct-0vzsr2nyraedl .	August 26, 2021

Change	Description	Link(s)
	Management | Advanced stack components | DNS (private) | Update The schema is updated with new parameters: **AliasTargetDnsName**, **AliasTargetHostedZoneId**, and **AliasTargetEvaluatedTargetHealth** to support "A" record to route traffic to AWS resource such as CloudFront distribution or an Amazon S3 bucket, by providing the DNSName and HostedZoneID associated with the AWS resource. ct-1d55pi44ff21u.	August 26, 2021
	Management | Advanced stack components | DNS (public) | Update The schema is updated with new parameters: **AliasTargetDnsName**, **AliasTargetHostedZoneId**, and **AliasTargetEvaluatedTargetHealth** to support "A" record to route traffic to AWS resource such as CloudFront distribution or an Amazon S3 bucket, by providing the DNSName and HostedZoneID associated with the AWS resource. ct-1hzofpphabs3i.	August 26, 2021
Updated CTs:	Deployment | Advanced stack components | RDS database stack | Create The **RDSDBEngine** parameter has a new value available: **mariadb**. ct-2z60dyvto9g6c.	August 12, 2021
New CTs:	Deployment | Advanced stack components | Identity and Access Management (IAM) | Create SAML identity provider ct-3hox8uwjgze1f.	July 29, 2021
	Management | Advanced stack components | Identity and Access Management (IAM) | Delete SAML identity provider ct-01zl37gmuk4q2.	July 29, 2021
	Management | Advanced stack components | Identity and Access Management (IAM) | Update SAML identity provider ct-379uwo67vbvng.	July 29, 2021

Change	Description	Link(s)
New CTs:	Deployment | Advanced stack components | VPNGateway | Create ct-0qbikxr9okwvy.	July 15, 2021
	Management | Advanced stack components | AMI | Create from Auto Scaling group ct-3e3prksxmdhw8.	July 15, 2021
	Management | Advanced stack components | EBS Volume | Attach ct-34jldf2qihaic.	July 15, 2021
	Management | Advanced stack components | EBS Volume | Detach ct-2d55p1d7z6w3d.	July 15, 2021
	Managed firewall, Outbound (Palo Alto): Create allow list ct-309eozh6lpkr8.	July 15, 2021
	Managed firewall, Outbound (Palo Alto): Delete allow list ct-2fzh1wckpl7f5.	July 15, 2021
	Managed firewall, Outbound (Palo Alto): Create security policy ct-281dpwh9tqnan.	July 15, 2021
	Managed firewall, Outbound (Palo Alto): Delete security policy ct-1taxucdyi84iy.	July 15, 2021
	Managed firewall, Outbound (Palo Alto): Update security policy ct-0mss4i7neuj7f.	July 15, 2021
Updated CTs:	Management | Managed firewall | Outbound (Palo Alto) | Add URLs and Management | Managed firewall | Outbound (Palo Alto) | Remove URLs. New schemas. ct-2b9q8339bj2sa and ct-2mf36chtp1ejh.	July 15, 2021
	Management | AWS service | Self-provisioned service | Add. New parameter, **SAMLProviders**. ct-3qe6io8t6jtny.	July 15, 2021

Change	Description	Link(s)
	Management | Advanced stack components | AMI | Encrypt. Note warning not to try to encrypt AMIs that are already encrypted. ct-3u9yd8jznb2zd.	July 15, 2021

AWS Glossary

For the latest AWS terminology, see the AWS glossary in the *AWS Glossary Reference*.