亚马逊云科技

API Reference

Amazon Secrets Manager

API Version 2017-10-17

Amazon Secrets Manager: API Reference

Table of Contents

Welcome

Amazon Secrets Manager provides a service to enable you to store, manage, and retrieve, secrets.

This guide provides descriptions of the Secrets Manager API. For more information about using this service, see the Amazon Secrets Manager User Guide.

API Version

This version of the Secrets Manager API Reference documents the Secrets Manager API version 2017-10-17.

Although you can make direct calls to the Secrets Manager HTTPS Query API, we recommend that you use one of the SDKs instead. The SDK performs many useful tasks you otherwise must perform manually. For example, the SDKs automatically sign your requests and convert responses into a structure syntactically appropriate to your language.

For SDKs, see:

- C++
- Java
- PHP
- Python
- Ruby
- .NET
- Node.js
- Go

Making HTTPS query requests

The Query API for Amazon Secrets Manager lets you call service operations. Query API requests are HTTPS requests that must contain an `Action` parameter to indicate the operation to be performed. Amazon Secrets Manager supports GET and POST requests for all operations. The API doesn't require you to use GET for some operations and POST for others. However, GET requests are subject to the limitation size of a URL. Although this limit depends on the browser, a typical limit is 2048 bytes. Therefore, for Query API requests that require larger sizes, you must use a POST request.

For a list of endpoints, see Amazon Secrets Manager endpoints.

The API returns the response in an XML document. For details about the response, see the individual API description pages in the Amazon Organizations API reference.

Because the Query API returns sensitive information such as security credentials, you must use HTTPS to encrypt all API requests.

When you send HTTP requests to Amazon, you must sign the requests so Amazon can identify the sender. You must use Signature Version 4. If you have an existing application that uses Signature Version 2, you must update it to use Signature Version 4.

You sign requests with your Amazon access key, which consists of an access key ID and a secret access key. We strongly recommend you don't create an access key for your root user. Anyone who has the access key for your root user has unrestricted access to all the resources in your account. Instead, create an access key for an IAM user with permissions required for the task at hand. As another option, use Amazon Security Token Service to generate temporary security credentials, and use those credentials to sign requests.

For more information, see the following:

- Amazon Security Credentials. Provides general information about the types of credentials you can use to access Amazon.
- IAM Best Practices. Offers suggestions for using the IAM service to help secure your Amazon resources, including those in Secrets Manager.
- Temporary Credentials. Describes how to create and use temporary security credentials.

When you use the Amazon Command Line Interface (Amazon CLI) or one of the Amazon SDKs to make requests to Amazon, these tools automatically sign the requests for you with the access key that you specify when you configure the tools.

The JSON that Amazon Secrets Manager expects as your request parameters and the service returns as a response to HTTP query requests contain single, long strings without line breaks or white space formatting. The JSON shown in the examples in this guide displays the code formatted with both line breaks and white space to improve readability. When example input parameters can also cause long strings extending beyond the screen, you can insert line breaks to enhance readability. You should always submit the input as a single JSON text string.

Support and Feedback for Amazon Secrets Manager

We welcome your feedback. Send your comments to awssecretsmanager-feedback@amazon.com, or post your feedback and questions in the Amazon Secrets Manager Discussion Forum. For more information about the Amazon Discussion Forums, see Forums Help.

Logging API Requests

Amazon Secrets Manager supports Amazon CloudTrail, a service that records Amazon API calls for your Amazon account and delivers log files to an Amazon S3 bucket. By using information that's collected by Amazon CloudTrail, you can determine the requests successfully made to Secrets Manager, who made the request, when it was made, and so on. For more about Amazon Secrets Manager and support for Amazon CloudTrail, see Logging Amazon Secrets Manager Events with Amazon CloudTrail in the *Amazon Secrets Manager User Guide*. To learn more about CloudTrail, including enabling it and find your log files, see the Amazon CloudTrail User Guide.

This document was last published on July 3, 2025.

Actions

The following actions are supported:

- BatchGetSecretValue
- CancelRotateSecret
- CreateSecret
- DeleteResourcePolicy
- DeleteSecret
- DescribeSecret
- GetRandomPassword
- GetResourcePolicy
- GetSecretValue
- ListSecrets
- ListSecretVersionIds
- PutResourcePolicy
- PutSecretValue
- RemoveRegionsFromReplication
- ReplicateSecretToRegions
- RestoreSecret
- RotateSecret
- StopReplicationToReplica
- TagResource
- UntagResource
- UpdateSecret
- UpdateSecretVersionStage
- ValidateResourcePolicy

BatchGetSecretValue

Retrieves the contents of the encrypted fields `SecretString` or `SecretBinary` for up to 20 secrets. To retrieve a single secret, call GetSecretValue.

To choose which secrets to retrieve, you can specify a list of secrets by name or ARN, or you can use filters. If Secrets Manager encounters errors such as `AccessDeniedException` while attempting to retrieve any of the secrets, you can see the errors in `Errors` in the response.

Secrets Manager generates CloudTrail `GetSecretValue` log entries for each secret you request when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:BatchGetSecretValue`, and you must have `secretsmanager:GetSecretValue` for each secret. If you use filters, you must also have `secretsmanager:ListSecrets`. If the secrets are encrypted using customer-managed keys instead of the Amazon managed key `aws/secretsmanager`, then you also need `kms:Decrypt` permissions for the keys. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "Filters": [
      {
         "Key": "string",
         "Values": [ "string" ]
      }
   ],
   "MaxResults": number,
   "NextToken": "string",
   "SecretIdList": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

Filters

The filters to choose which secrets to retrieve. You must include `Filters` or `SecretIdList`, but not both.

Type: Array of Filter objects

Array Members: Maximum number of 10 items.

Required: No

MaxResults

The number of results to include in the response.

If there are more results available, in the response, Secrets Manager includes `NextToken`. To get the next results, call `BatchGetSecretValue` again with the value from `NextToken`. To use this parameter, you must also use the `Filters` parameter.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 20.

Required: No

NextToken

A token that indicates where the output should continue from, if a previous call did not show all results. To get the next results, call `BatchGetSecretValue` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Required: No

SecretIdList

The ARN or names of the secrets to retrieve. You must include `Filters` or `SecretIdList`, but not both.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

Response Syntax

```
{
   "Errors": [
      {
         "ErrorCode": "string",
         "Message": "string",
         "SecretId": "string"
      }
   ],
   "NextToken": "string",
   "SecretValues": [
      {
         "ARN": "string",
         "CreatedDate": number,
         "Name": "string",
         "SecretBinary": blob,
         "SecretString": "string",
         "VersionId": "string",
         "VersionStages": [ "string" ]
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Errors

A list of errors Secrets Manager encountered while attempting to retrieve individual secrets.

Type: Array of APIErrorType objects

NextToken

Secrets Manager includes this value if there's more output available than what is included in the current response. This can occur even when the response includes no values at all, such as when

you ask for a filtered view of a long list. To get the next results, call `BatchGetSecretValue` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

SecretValues

A list of secret values.

Type: Array of SecretValueEntry objects

Errors

For information about the errors that are common to all actions, see Common Errors.

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidNextTokenException

The `NextToken` value is invalid.

HTTP Status Code: 400

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to retrieve the secret values for a group of secrets listed by name. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.BatchGetSecretValue
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretIdList": ["MySecret1", "MySecret2", "MySecret3"]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

  {
    "Errors":[],
    "SecretValues":
      [
        {
          "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret1-
a1b2c3",
          "CreatedDate":1.700591229801E9,
          "Name":"MySecret1",
          "SecretString":"{\"username\":\"diego_ramirez\",\"password\":\"EXAMPLE-
PASSWORD\",\"engine\":\"mysql\",\"host\":\"secretsmanagertutorial.cluster.us-
west-2.rds.amazonaws.com\",\"port\":3306,\"dbClusterIdentifier\":
\"secretsmanagertutorial\"}",
          "VersionId":"a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa",
          "VersionStages": ["AWSCURRENT"]
        },
        {
          "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret2-
a1b2c3",
          "CreatedDate":1.699911394105E9,
          "Name":"MySecret2",
          "SecretString":"{\"username\":\"akua_mansa\",\"password\":\"EXAMPLE-PASSWORD
\"",
          "VersionId":"a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
          "VersionStages":["AWSCURRENT"]
        },
        {
          "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret3-
a1b2c3",
          "CreatedDate":1.699911394105E9,
          "Name":"MySecret3",
          "SecretString":"{\"username\":\"jie_liu\",\"password\":\"EXAMPLE-PASSWORD\"",
          "VersionId":"a1b2c3d4-5678-90ab-cdef-EXAMPLEccccc",
          "VersionStages":["AWSCURRENT"]
        }
```

```
    ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

CancelRotateSecret

Turns off automatic rotation, and if a rotation is currently in progress, cancels the rotation.

If you cancel a rotation in progress, it can leave the `VersionStage` labels in an unexpected state. You might need to remove the staging label `AWSPENDING` from the partially created version. You also need to determine whether to roll back to the previous version of the secret by moving the staging label `AWSCURRENT` to the version that has `AWSPENDING`. To determine which version has a specific staging label, call ListSecretVersionIds. Then use UpdateSecretVersionStage to change staging labels. For more information, see How rotation works.

To turn on automatic rotation again, call RotateSecret.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:CancelRotateSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "VersionId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

VersionId

The unique identifier of the version of the secret created during the rotation. This version might not be complete, and should be evaluated for possible deletion. We recommend that you remove the `VersionStage` value `AWSPENDING` from this version so that Secrets Manager can delete it. Failing to clean up a cancelled rotation can block you from starting future rotations.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to cancel rotation for a secret.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.CancelRotateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2

- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

CreateSecret

Creates a new secret. A *secret* can be a password, a set of credentials such as a user name and password, an OAuth token, or other secret information that you store in an encrypted form in Secrets Manager. The secret also includes the connection information to access a database or other service, which Secrets Manager doesn't encrypt. A secret in Secrets Manager consists of both the protected secret data and the important information needed to manage the secret.

For secrets that use *managed rotation*, you need to create the secret through the managing service. For more information, see Secrets Manager secrets managed by other Amazon services.

For information about creating a secret in the console, see Create a secret.

To create a secret, you can provide the secret value to be encrypted in either the `SecretString` parameter or the `SecretBinary` parameter, but not both. If you include `SecretString` or `SecretBinary` then Secrets Manager creates an initial secret version and automatically attaches the staging label `AWSCURRENT` to it.

For database credentials you want to rotate, for Secrets Manager to be able to rotate the secret, you must make sure the JSON you store in the `SecretString` matches the JSON structure of a database secret.

If you don't specify an Amazon KMS encryption key, Secrets Manager uses the Amazon managed key `aws/secretsmanager`. If this key doesn't already exist in your account, then Secrets Manager creates it for you automatically. All users and roles in the Amazon account automatically have access to use `aws/secretsmanager`. Creating `aws/secretsmanager` can result in a one-time significant delay in returning the result.

If the secret is in a different Amazon account from the credentials calling the API, then you can't use `aws/secretsmanager` to encrypt the secret, and you must create and use a customer managed Amazon KMS key.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters except `SecretBinary` or `SecretString` because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:CreateSecret`. If you include tags in the secret, you also need `secretsmanager:TagResource`. To add replica Regions, you must also have

`secretsmanager:ReplicateSecretToRegions`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

To encrypt the secret with a KMS key other than `aws/secretsmanager`, you need `kms:GenerateDataKey` and `kms:Decrypt` permission to the key.

> **Important**
>
> When you enter commands in a command shell, there is a risk of the command history being accessed or utilities having access to your command parameters. This is a concern if the command includes the value of a secret. Learn how to Mitigate the risks of using command-line tools to store Amazon Secrets Manager secrets.

Request Syntax

```
{
   "AddReplicaRegions": [
      {
         "KmsKeyId": "string",
         "Region": "string"
      }
   ],
   "ClientRequestToken": "string",
   "Description": "string",
   "ForceOverwriteReplicaSecret": boolean,
   "KmsKeyId": "string",
   "Name": "string",
   "SecretBinary": blob,
   "SecretString": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

AddReplicaRegions

A list of Regions and Amazon KMS keys to replicate secrets.

Type: Array of ReplicaRegionType objects

Array Members: Minimum number of 1 item.

Required: No

ClientRequestToken

If you include `SecretString` or `SecretBinary`, then Secrets Manager creates an initial version for the secret, and this parameter specifies the unique identifier for the new version.

> **Note**
>
> If you use the Amazon CLI or one of the Amazon SDKs to call this operation, then you can leave this parameter empty. The CLI or SDK generates a random UUID for you and includes it as the value for this parameter in the request.

If you generate a raw HTTP request to the Secrets Manager service endpoint, then you must generate a `ClientRequestToken` and include it in the request.

This value helps ensure idempotency. Secrets Manager uses this value to prevent the accidental creation of duplicate versions if there are failures and retries during a rotation. We recommend that you generate a UUID-type value to ensure uniqueness of your versions within the specified secret.

- If the `ClientRequestToken` value isn't already associated with a version of the secret then a new version of the secret is created.
- If a version with this value already exists and the version `SecretString` and `SecretBinary` values are the same as those in the request, then the request is ignored.
- If a version with this value already exists and that version's `SecretString` and `SecretBinary` values are different from those in the request, then the request fails because you cannot modify an existing version. Instead, use PutSecretValue to create a new version.

This value becomes the `VersionId` of the new version.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

Description

The description of the secret.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

ForceOverwriteReplicaSecret

Specifies whether to overwrite a secret with the same name in the destination Region. By default, secrets aren't overwritten.

Type: Boolean

Required: No

KmsKeyId

The ARN, key ID, or alias of the Amazon KMS key that Secrets Manager uses to encrypt the secret value in the secret. An alias is always prefixed by `alias/`, for example `alias/aws/secretsmanager`. For more information, see About aliases.

To use a Amazon KMS key in a different account, use the key ARN or the alias ARN.

If you don't specify this value, then Secrets Manager uses the key `aws/secretsmanager`. If that key doesn't yet exist, then Secrets Manager creates it for you automatically the first time it encrypts the secret value.

If the secret is in a different Amazon account from the credentials calling the API, then you can't use `aws/secretsmanager` to encrypt the secret, and you must create and use a customer managed Amazon KMS key.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

Name

The name of the new secret.

The secret name can contain ASCII letters, numbers, and the following characters: /_+=.@-

Do not end your secret name with a hyphen followed by six characters. If you do so, you risk confusion and unexpected results when searching for a secret by partial ARN. Secrets Manager automatically adds a hyphen and six random characters after the secret name at the end of the ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 512.

Required: Yes

SecretBinary

The binary data to encrypt and store in the new version of the secret. We recommend that you store your binary data in a file and then pass the contents of the file as a parameter.

Either `SecretString` or `SecretBinary` must have a value, but not both.

This parameter is not available in the Secrets Manager console.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

SecretString

The text data to encrypt and store in this new version of the secret. We recommend you use a JSON structure of key/value pairs for your secret value.

Either `SecretString` or `SecretBinary` must have a value, but not both.

If you create a secret by using the Secrets Manager console then Secrets Manager puts the protected secret text in only the `SecretString` parameter. The Secrets Manager console stores the information as a JSON structure of key/value pairs that a Lambda rotation function can parse.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

Tags

A list of tags to attach to the secret. Each tag is a key and value pair of strings in a JSON text string, for example:

```
[{"Key":"CostCenter","Value":"12345"},
{"Key":"environment","Value":"production"}]
```

Secrets Manager tag key names are case sensitive. A tag with the key "ABC" is a different tag from one with key "abc".

If you check tags in permissions policies as part of your security strategy, then adding or removing a tag can change permissions. If the completion of this operation would result in you losing your permissions for this secret, then Secrets Manager blocks the operation and returns an `Access Denied` error. For more information, see Control access to secrets using tags and Limit access to identities with tags that match secrets' tags.

For information about how to format a JSON parameter for the various command line tool environments, see Using JSON for Parameters. If your command-line tool or SDK requires quotation marks around the parameter, you should use single quotes to avoid confusion with the double quotes required in the JSON text.

For tag quotas and naming restrictions, see Service quotas for Tagging in the *Amazon General Reference guide*.

Type: Array of Tag objects

Required: No

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "ReplicationStatus": [
      {
         "KmsKeyId": "string",
         "LastAccessedDate": number,
         "Region": "string",
         "Status": "string",
         "StatusMessage": "string"
      }
   ],
   "VersionId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the new secret. The ARN includes the name of the secret followed by six random characters. This ensures that if you create a new secret with the same name as a deleted secret, then users with access to the old secret don't get access to the new secret because the ARNs are different.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the new secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

ReplicationStatus

A list of the replicas of this secret and their status:

- `Failed`, which indicates that the replica was not created.
- `InProgress`, which indicates that Secrets Manager is in the process of creating the replica.
- `InSync`, which indicates that the replica was created.

Type: Array of ReplicationStatusType objects

VersionId

The unique identifier associated with the version of the new secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Errors

For information about the errors that are common to all actions, see Common Errors.

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

EncryptionFailure

Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the KMS key is available, enabled, and not in an invalid state. For more information, see Key state: Effect on your KMS key.

HTTP Status Code: 400

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

LimitExceededException

The request failed because it would exceed one of the Secrets Manager quotas.

HTTP Status Code: 400

MalformedPolicyDocumentException

The resource policy has syntax errors.

HTTP Status Code: 400

PreconditionNotMetException

The request failed because you did not complete all the prerequisite steps.

HTTP Status Code: 400

ResourceExistsException

A resource with the ID you requested already exists.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to create a secret. Secrets Manager retrieves the credentials stored in the encrypted secret value from a file on disk named mycreds.json. For an example of mycreds.json, see Creating a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.CreateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "Name": "MyTestDatabaseSecret",
  "Description": "My test database secret created with the CLI",
  "SecretString": "{\"username\":\"david\",\"password\":\"EXAMPLE-PASSWORD\"}",
 "ClientRequestToken": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>
```

```
{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret",
  "VersionId": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

DeleteResourcePolicy

Deletes the resource-based permission policy attached to the secret. To attach a policy to a secret, use PutResourcePolicy.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:DeleteResourcePolicy`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret to delete the attached resource-based policy for.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
```

```
    "ARN": "string",
    "Name": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret that the resource-based policy was deleted for.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret that the resource-based policy was deleted for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 512.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to delete the resource-based policy that's attached to a secret.The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.DeleteResourcePolicy
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

DeleteSecret

Deletes a secret and all of its versions. You can specify a recovery window during which you can restore the secret. The minimum recovery window is 7 days. The default recovery window is 30 days. Secrets Manager attaches a `DeletionDate` stamp to the secret that specifies the end of the recovery window. At the end of the recovery window, Secrets Manager deletes the secret permanently.

You can't delete a primary secret that is replicated to other Regions. You must first delete the replicas using RemoveRegionsFromReplication, and then delete the primary secret. When you delete a replica, it is deleted immediately.

You can't directly delete a version of a secret. Instead, you remove all staging labels from the version using UpdateSecretVersionStage. This marks the version as deprecated, and then Secrets Manager can automatically delete the version in the background.

To determine whether an application still uses a secret, you can create an Amazon CloudWatch alarm to alert you to any attempts to access a secret during the recovery window. For more information, see Monitor secrets scheduled for deletion.

Secrets Manager performs the permanent secret deletion at the end of the waiting period as a background task with low priority. There is no guarantee of a specific time after the recovery window for the permanent delete to occur.

At any time before recovery window ends, you can use RestoreSecret to remove the `DeletionDate` and cancel the deletion of the secret.

When a secret is scheduled for deletion, you cannot retrieve the secret value. You must first cancel the deletion with RestoreSecret and then you can retrieve the secret.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:DeleteSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
```

```
    "ForceDeleteWithoutRecovery": boolean,
    "RecoveryWindowInDays": number,
    "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ForceDeleteWithoutRecovery

Specifies whether to delete the secret without any recovery window. You can't use both this parameter and `RecoveryWindowInDays` in the same call. If you don't use either, then by default Secrets Manager uses a 30 day recovery window.

Secrets Manager performs the actual deletion with an asynchronous background process, so there might be a short delay before the secret is permanently deleted. If you delete a secret and then immediately create a secret with the same name, use appropriate back off and retry logic.

If you forcibly delete an already deleted or nonexistent secret, the operation does not return `ResourceNotFoundException`.

> **Important**
>
> Use this parameter with caution. This parameter causes the operation to skip the normal recovery window before the permanent deletion that Secrets Manager would normally impose with the `RecoveryWindowInDays` parameter. If you delete a secret with the `ForceDeleteWithoutRecovery` parameter, then you have no opportunity to recover the secret. You lose the secret permanently.

Type: Boolean

Required: No

RecoveryWindowInDays

The number of days from 7 to 30 that Secrets Manager waits before permanently deleting the secret. You can't use both this parameter and `ForceDeleteWithoutRecovery` in the same call. If you don't use either, then by default Secrets Manager uses a 30 day recovery window.

Type: Long

Required: No

SecretId

The ARN or name of the secret to delete.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "DeletionDate": number,
   "Name": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

DeletionDate

The date and time after which this secret Secrets Manager can permanently delete this secret, and it can no longer be restored. This value is the date and time of the delete request plus the number of days in `RecoveryWindowInDays`.

Type: Timestamp

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to delete a secret with a recovery window of 7 days. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.DeleteSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "RecoveryWindowInDays": 7
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "DeletionDate":1.524085349095E9,
```

```
  "Name":"MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

DescribeSecret

Retrieves the details of a secret. It does not include the encrypted secret value. Secrets Manager only returns fields that have a value in the response.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:DescribeSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
```

```
    "CreatedDate": number,
    "DeletedDate": number,
    "Description": "string",
    "KmsKeyId": "string",
    "LastAccessedDate": number,
    "LastChangedDate": number,
    "LastRotatedDate": number,
    "Name": "string",
    "NextRotationDate": number,
    "OwningService": "string",
    "PrimaryRegion": "string",
    "ReplicationStatus": [
       {
          "KmsKeyId": "string",
          "LastAccessedDate": number,
          "Region": "string",
          "Status": "string",
          "StatusMessage": "string"
       }
    ],
    "RotationEnabled": boolean,
    "RotationLambdaARN": "string",
    "RotationRules": {
       "AutomaticallyAfterDays": number,
       "Duration": "string",
       "ScheduleExpression": "string"
    },
    "Tags": [
       {
          "Key": "string",
          "Value": "string"
       }
    ],
    "VersionIdsToStages": {
       "string" : [ "string" ]
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

CreatedDate

The date the secret was created.

Type: Timestamp

DeletedDate

The date the secret is scheduled for deletion. If it is not scheduled for deletion, this field is omitted. When you delete a secret, Secrets Manager requires a recovery window of at least 7 days before deleting the secret. Some time after the deleted date, Secrets Manager deletes the secret, including all of its versions.

If a secret is scheduled for deletion, then its details, including the encrypted secret value, is not accessible. To cancel a scheduled deletion and restore access to the secret, use RestoreSecret.

Type: Timestamp

Description

The description of the secret.

Type: String

Length Constraints: Maximum length of 2048.

KmsKeyId

The key ID or alias ARN of the Amazon KMS key that Secrets Manager uses to encrypt the secret value. If the secret is encrypted with the Amazon managed key `aws/secretsmanager`, this field is omitted. Secrets created using the console use an Amazon KMS key ID.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

LastAccessedDate

The date that the secret was last accessed in the Region. This field is omitted if the secret has never been retrieved in the Region.

Type: Timestamp

LastChangedDate

The last date and time that this secret was modified in any way.

Type: Timestamp

LastRotatedDate

The last date and time that Secrets Manager rotated the secret. If the secret isn't configured for rotation or rotation has been disabled, Secrets Manager returns null.

Type: Timestamp

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

NextRotationDate

The next rotation is scheduled to occur on or before this date. If the secret isn't configured for rotation or rotation has been disabled, Secrets Manager returns null. If rotation fails, Secrets Manager retries the entire rotation process multiple times. If rotation is unsuccessful, this date may be in the past.

This date represents the latest date that rotation will occur, but it is not an approximate rotation date. In some cases, for example if you turn off automatic rotation and then turn it back on, the next rotation may occur much sooner than this date.

Type: Timestamp

OwningService

The ID of the service that created this secret. For more information, see Secrets managed by other Amazon services.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

PrimaryRegion

The Region the secret is in. If a secret is replicated to other Regions, the replicas are listed in `ReplicationStatus`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([a-z]+-)+\d+$`

ReplicationStatus

A list of the replicas of this secret and their status:

- `Failed`, which indicates that the replica was not created.
- `InProgress`, which indicates that Secrets Manager is in the process of creating the replica.
- `InSync`, which indicates that the replica was created.

Type: Array of ReplicationStatusType objects

RotationEnabled

Specifies whether automatic rotation is turned on for this secret. If the secret has never been configured for rotation, Secrets Manager returns null.

To turn on rotation, use RotateSecret. To turn off rotation, use CancelRotateSecret.

Type: Boolean

RotationLambdaARN

The ARN of the Lambda function that Secrets Manager invokes to rotate the secret.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

RotationRules

The rotation schedule and Lambda function for this secret. If the secret previously had rotation turned on, but it is now turned off, this field shows the previous rotation schedule and rotation function. If the secret never had rotation turned on, this field is omitted.

Type: RotationRulesType object

Tags

The list of tags attached to the secret. To add tags to a secret, use TagResource. To remove tags, use UntagResource.

Type: Array of Tag objects

VersionIdsToStages

A list of the versions of the secret that have staging labels attached. Versions that don't have staging labels are considered deprecated and Secrets Manager can delete them.

Secrets Manager uses staging labels to indicate the status of a secret version during rotation. The three staging labels for rotation are:

- `AWSCURRENT`, which indicates the current version of the secret.
- `AWSPENDING`, which indicates the version of the secret that contains new secret information that will become the next current version when rotation finishes.

  During rotation, Secrets Manager creates an `AWSPENDING` version ID before creating the new secret version. To check if a secret version exists, call GetSecretValue.
- `AWSPREVIOUS`, which indicates the previous current version of the secret. You can use this as the *last known good* version.

For more information about rotation and staging labels, see How rotation works.

Type: String to array of strings map

Key Length Constraints: Minimum length of 32. Maximum length of 64.

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to get the details about a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.DescribeSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
```

```
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "Description": "My test database secret created with the CLI",
  "LastChangedDate": 1523477145.729,
  "LastAccessedDate": 1606269226,
  "RotationEnabled": true,
  "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestRotationLambda",
  "RotationRules": {
    "AutomaticallyAfterDays": 14,
    "ScheduleExpression": "cron(0 16 1,15 * ? *)",
    "Duration": "2h"
  },
  "LastRotatedDate": 1525747253.72,
  "NextRotationDate": 1665165599000,
  "Tags": [
    {
      "Key": "SecondTag",
      "Value": "AnotherValue"
    },
    {
      "Key": "FirstTag",
      "Value": "SomeValue"
    }
  ],
  "VersionIdsToStages": {
    "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1": [
      "AWSPREVIOUS"
    ],
    "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2": [
      "AWSCURRENT"
    ]
  }
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

GetRandomPassword

Generates a random password. We recommend that you specify the maximum length and include every character type that the system you are generating a password for can support. By default, Secrets Manager uses uppercase and lowercase letters, numbers, and the following characters in passwords: `!\"#$%&'()*+,-./:;<=>?@[\\]^_`{|}~`

Secrets Manager generates a CloudTrail log entry when you call this action.

Required permissions: `secretsmanager:GetRandomPassword`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "ExcludeCharacters": "string",
   "ExcludeLowercase": boolean,
   "ExcludeNumbers": boolean,
   "ExcludePunctuation": boolean,
   "ExcludeUppercase": boolean,
   "IncludeSpace": boolean,
   "PasswordLength": number,
   "RequireEachIncludedType": boolean
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ExcludeCharacters

A string of the characters that you don't want in the password.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 4096.

Required: No

ExcludeLowercase

Specifies whether to exclude lowercase letters from the password. If you don't include this switch, the password can contain lowercase letters.

Type: Boolean

Required: No

ExcludeNumbers

Specifies whether to exclude numbers from the password. If you don't include this switch, the password can contain numbers.

Type: Boolean

Required: No

ExcludePunctuation

Specifies whether to exclude the following punctuation characters from the password: `! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [ \ ] ^ _ ` { | } ~`. If you don't include this switch, the password can contain punctuation.

Type: Boolean

Required: No

ExcludeUppercase

Specifies whether to exclude uppercase letters from the password. If you don't include this switch, the password can contain uppercase letters.

Type: Boolean

Required: No

IncludeSpace

Specifies whether to include the space character. If you include this switch, the password can contain space characters.

Type: Boolean

Required: No

PasswordLength

The length of the password. If you don't include this parameter, the default length is 32 characters.

Type: Long

Valid Range: Minimum value of 1. Maximum value of 4096.

Required: No

RequireEachIncludedType

Specifies whether to include at least one upper and lowercase letter, one number, and one punctuation. If you don't include this switch, the password contains at least one of every character type.

Type: Boolean

Required: No

Response Syntax

```
{
   "RandomPassword": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

RandomPassword

A string with the password.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 4096.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

Examples

Example

The following example shows how to request a randomly generated password of 20 characters.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.GetRandomPassword
Content-Type: application/x-amz-json-1.1
```

```
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "PasswordLength": 20
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "RandomPassword":"N+Z43a,>vx7j O8^*<8i3"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

GetResourcePolicy

Retrieves the JSON text of the resource-based policy document attached to the secret. For more information about permissions policies attached to a secret, see Permissions policies attached to a secret.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:GetResourcePolicy`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret to retrieve the attached resource-based policy for.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
```

```
    "ARN": "string",
    "Name": "string",
    "ResourcePolicy": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret that the resource-based policy was retrieved for.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret that the resource-based policy was retrieved for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 512.

ResourcePolicy

A JSON-formatted string that contains the permissions policy attached to the secret. For more information about permissions policies, see Authentication and access control for Secrets Manager.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20480.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to retrieve the resource-based policy attached to a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.GetResourcePolicy
Content-Type: application/x-amz-json-1.1
```

```
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "ResourcePolicy": "{\"Version\":\"2012-10-17\",\"Statement\":{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":[\"arn:aws:iam::111122223333:root\",
\"arn:aws:iam::444455556666:root\"]},\"Action\":[\"secretsmanager:GetSecretValue\"],
\"Resource\":\"*\"}}"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3

- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

GetSecretValue

Retrieves the contents of the encrypted fields `SecretString` or `SecretBinary` from the specified version of a secret, whichever contains content.

To retrieve the values for a group of secrets, call BatchGetSecretValue.

We recommend that you cache your secret values by using client-side caching. Caching secrets improves speed and reduces your costs. For more information, see Cache secrets for your applications.

To retrieve the previous version of a secret, use `VersionStage` and specify AWSPREVIOUS. To revert to the previous version of a secret, call UpdateSecretVersionStage.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:GetSecretValue`. If the secret is encrypted using a customer-managed key instead of the Amazon managed key `aws/secretsmanager`, then you also need `kms:Decrypt` permissions for that key. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string",
   "VersionId": "string",
   "VersionStage": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret to retrieve. To retrieve a secret from another account, you must use an ARN.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

VersionId

The unique identifier of the version of the secret to retrieve. If you include both this parameter and `VersionStage`, the two parameters must refer to the same secret version. If you don't specify either a `VersionStage` or `VersionId`, then Secrets Manager returns the `AWSCURRENT` version.

This value is typically a UUID-type value with 32 hexadecimal digits.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

VersionStage

The staging label of the version of the secret to retrieve.

Secrets Manager uses staging labels to keep track of different versions during the rotation process. If you include both this parameter and `VersionId`, the two parameters must refer to the same secret version. If you don't specify either a `VersionStage` or `VersionId`, Secrets Manager returns the `AWSCURRENT` version.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Response Syntax

```
{
```

```
    "ARN": "string",
    "CreatedDate": number,
    "Name": "string",
    "SecretBinary": blob,
    "SecretString": "string",
    "VersionId": "string",
    "VersionStages": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

CreatedDate

The date and time that this version of the secret was created. If you don't specify which version in `VersionId` or `VersionStage`, then Secrets Manager uses the `AWSCURRENT` version.

Type: Timestamp

Name

The friendly name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

SecretBinary

The decrypted secret value, if the secret value was originally provided as binary data in the form of a byte array. When you retrieve a `SecretBinary` using the HTTP API, the Python SDK, or the Amazon CLI, the value is Base64-encoded. Otherwise, it is not encoded.

If the secret was created by using the Secrets Manager console, or if the secret value was originally provided as a string, then this field is omitted. The secret value appears in `SecretString` instead.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 65536.

SecretString

The decrypted secret value, if the secret value was originally provided as a string or through the Secrets Manager console.

If this secret was created by using the console, then Secrets Manager stores the information as a JSON structure of key/value pairs.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65536.

VersionId

The unique identifier of this version of the secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

VersionStages

A list of all of the staging labels currently attached to this version of the secret.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to retrieve the secret value from a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.GetSecretValue
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "CreatedDate":1.523477145713E9,
  "Name":"MyTestDatabaseSecret",
  "SecretString":"{\n  \"username\":\"david\",\n  \"password\":\"EXAMPLE-PASSWORD
\"\n}\n",
  "VersionId":"EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
```

```
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

ListSecrets

Lists the secrets that are stored by Secrets Manager in the Amazon account, not including secrets that are marked for deletion. To see secrets marked for deletion, use the Secrets Manager console.

All Secrets Manager operations are eventually consistent. ListSecrets might not reflect changes from the last five minutes. You can get more recent information for a specific secret by calling DescribeSecret.

To list the versions of a secret, use ListSecretVersionIds.

To retrieve the values for the secrets, call BatchGetSecretValue or GetSecretValue.

For information about finding secrets in the console, see Find secrets in Secrets Manager.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:ListSecrets`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "Filters": [
      {
         "Key": "string",
         "Values": [ "string" ]
      }
   ],
   "IncludePlannedDeletion": boolean,
   "MaxResults": number,
   "NextToken": "string",
   "SortOrder": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

Filters

The filters to apply to the list of secrets.

Type: Array of Filter objects

Array Members: Maximum number of 10 items.

Required: No

IncludePlannedDeletion

Specifies whether to include secrets scheduled for deletion. By default, secrets scheduled for deletion aren't included.

Type: Boolean

Required: No

MaxResults

The number of results to include in the response.

If there are more results available, in the response, Secrets Manager includes `NextToken`. To get the next results, call `ListSecrets` again with the value from `NextToken`.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

NextToken

A token that indicates where the output should continue from, if a previous call did not show all results. To get the next results, call `ListSecrets` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Required: No

SortOrder

Secrets are listed by `CreatedDate`.

Type: String

Valid Values: `asc | desc`

Required: No

Response Syntax

```
{
   "NextToken": "string",
   "SecretList": [
      {
         "ARN": "string",
         "CreatedDate": number,
         "DeletedDate": number,
         "Description": "string",
         "KmsKeyId": "string",
         "LastAccessedDate": number,
         "LastChangedDate": number,
         "LastRotatedDate": number,
         "Name": "string",
         "NextRotationDate": number,
         "OwningService": "string",
         "PrimaryRegion": "string",
         "RotationEnabled": boolean,
         "RotationLambdaARN": "string",
         "RotationRules": {
            "AutomaticallyAfterDays": number,
            "Duration": "string",
            "ScheduleExpression": "string"
         },
         "SecretVersionsToStages": {
            "string" : [ "string" ]
         },
         "Tags": [
            {
               "Key": "string",
               "Value": "string"
            }
```

```
            ]
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

NextToken

Secrets Manager includes this value if there's more output available than what is included in the current response. This can occur even when the response includes no values at all, such as when you ask for a filtered view of a long list. To get the next results, call `ListSecrets` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

SecretList

A list of the secrets in the account.

Type: Array of SecretListEntry objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidNextTokenException

The `NextToken` value is invalid.

HTTP Status Code: 400

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

Examples

Example

The following example shows how to list the secrets in the account. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.ListSecrets
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>
```

```
{}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "SecretList":[
    {
      "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
      "Description":"My test database secret",
      "LastChangedDate":1.523477145729E9,
      "Name":"MyTestDatabaseSecret",
      "SecretVersionsToStages":{
        "EXAMPLE2-90ab-cdef-fedc-ba987EXAMPLE":["AWSCURRENT"]
      }
    },
    {
      "ARN":"arn:aws:secretsmanager:us-
west-2:123456789012:secret:AnotherDatabaseSecret-d4e5f6",
      "Description":"Another secret created for a different database",
      "LastChangedDate":1.523482025685E9,
      "Name":"AnotherDatabaseSecret",
      "SecretVersionsToStages":{
        "EXAMPLE3-90ab-cdef-fedc-ba987EXAMPLE":["AWSCURRENT"]
      }
    }
  ]
}
```

Example

The following example shows how to list the secrets in the account that are tagged with `costcenter 12345`. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.ListSecrets
&Filter.1.Name=costcenter
&Filter.1.Value.1=12345
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "SecretList":[
    {
      "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
      "Description":"My test database secret",
      "LastChangedDate":1.523477145729E9,
      "Name":"MyTestDatabaseSecret",
      "SecretVersionsToStages":{
        "EXAMPLE2-90ab-cdef-fedc-ba987EXAMPLE":["AWSCURRENT"]
      }
    },
    {
      "ARN":"arn:aws:secretsmanager:us-
west-2:123456789012:secret:AnotherDatabaseSecret-d4e5f6",
      "Description":"Another secret created for a different database",
      "LastChangedDate":1.523482025685E9,
```

```
      "Name":"AnotherDatabaseSecret",
      "SecretVersionsToStages":{
        "EXAMPLE3-90ab-cdef-fedc-ba987EXAMPLE":["AWSCURRENT"]
      }
    }
  ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

ListSecretVersionIds

Lists the versions of a secret. Secrets Manager uses staging labels to indicate the different versions of a secret. For more information, see Secrets Manager concepts: Versions.

To list the secrets in the account, use ListSecrets.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:ListSecretVersionIds`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "IncludeDeprecated": boolean,
   "MaxResults": number,
   "NextToken": "string",
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

IncludeDeprecated

Specifies whether to include versions of secrets that don't have any staging labels attached to them. Versions without staging labels are considered deprecated and are subject to deletion by Secrets Manager. By default, versions without staging labels aren't included.

Type: Boolean

Required: No

MaxResults

The number of results to include in the response.

If there are more results available, in the response, Secrets Manager includes `NextToken`. To get the next results, call `ListSecretVersionIds` again with the value from `NextToken`.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

NextToken

A token that indicates where the output should continue from, if a previous call did not show all results. To get the next results, call `ListSecretVersionIds` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Required: No

SecretId

The ARN or name of the secret whose versions you want to list.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "NextToken": "string",
   "Versions": [
      {
         "CreatedDate": number,
         "KmsKeyIds": [ "string" ],
```

```
            "LastAccessedDate": number,
            "VersionId": "string",
            "VersionStages": [ "string" ]
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

NextToken

Secrets Manager includes this value if there's more output available than what is included in the current response. This can occur even when the response includes no values at all, such as when you ask for a filtered view of a long list. To get the next results, call `ListSecretVersionIds` again with this value.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Versions

A list of the versions of the secret.

Type: Array of SecretVersionsListEntry objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidNextTokenException

The `NextToken` value is invalid.

HTTP Status Code: 400

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to retrieve a list of the versions of a secret, including versions which have no staging labels attached. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
```

```
X-Amz-Target: secretsmanager.ListSecretVersionIds
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "IncludeDeprecated": true
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret",
  "Versions":[
    {
      "CreatedDate":1.523477145713E9,
      "VersionId":"EXAMPLE1-90ab-cdef-fedc-ba987SECRET1",
      "VersionStages":["AWSPREVIOUS"]
    },
    {
      "CreatedDate":1.523486221391E9,
      "VersionId":"EXAMPLE2-90ab-cdef-fedc-ba987SECRET2",
      "VersionStages":["AWSCURRENT"]
    },
    {
      "CreatedDate": 1.51197446236E9,
      "VersionId": "EXAMPLE3-90ab-cdef-fedc-ba987SECRET3"
    }
  ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

PutResourcePolicy

Attaches a resource-based permission policy to a secret. A resource-based policy is optional. For more information, see Authentication and access control for Secrets Manager

For information about attaching a policy in the console, see Attach a permissions policy to a secret.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:PutResourcePolicy`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "BlockPublicPolicy": boolean,
   "ResourcePolicy": "string",
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

BlockPublicPolicy

Specifies whether to block resource-based policies that allow broad access to the secret, for example those that use a wildcard for the principal. By default, public policies aren't blocked.

> **Important**
>
> Resource policy validation and the BlockPublicPolicy parameter help protect your resources by preventing public access from being granted through the resource policies that are directly attached to your secrets. In addition to using these features, carefully inspect the following policies to confirm that they do not grant public access:

- Identity-based policies attached to associated Amazon principals (for example, IAM roles)
- Resource-based policies attached to associated Amazon resources (for example, Amazon Key Management Service (Amazon KMS) keys)

To review permissions to your secrets, see Determine who has permissions to your secrets.

Type: Boolean

Required: No

ResourcePolicy

A JSON-formatted string for an Amazon resource-based policy. For example policies, see Permissions policy examples.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20480.

Required: Yes

SecretId

The ARN or name of the secret to attach the resource-based policy.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "Name": "string"
```

```
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 512.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

MalformedPolicyDocumentException

The resource policy has syntax errors.

HTTP Status Code: 400

PublicPolicyException

The `BlockPublicPolicy` parameter is set to true, and the resource policy did not prevent broad access to the secret.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to attach a resource-based policy to a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.PutResourcePolicy
Content-Type: application/x-amz-json-1.1
```

```
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "ResourcePolicy": "{\"Version\":\"2012-10-17\",\"Statement\":{\"Effect
\":\"Allow\",\"Principal\":{\"AWS\":[\"arn:aws:iam::111122223333:root\",
\"arn:aws:iam::444455556666:root\"]},\"Action\":[\"secretsmanager:GetSecretValue\"],
\"Resource\":\"*\"}}"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3

- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

PutSecretValue

Creates a new version of your secret by creating a new encrypted value and attaching it to the secret. version can contain a new `SecretString` value or a new `SecretBinary` value.

Do not call `PutSecretValue` at a sustained rate of more than once every 10 minutes. When you update the secret value, Secrets Manager creates a new version of the secret. Secrets Manager keeps 100 of the most recent versions, but it keeps *all* secret versions created in the last 24 hours. If you call `PutSecretValue` more than once every 10 minutes, you will create more versions than Secrets Manager removes, and you will reach the quota for secret versions.

You can specify the staging labels to attach to the new version in `VersionStages`. If you don't include `VersionStages`, then Secrets Manager automatically moves the staging label `AWSCURRENT` to this version. If this operation creates the first version for the secret, then Secrets Manager automatically attaches the staging label `AWSCURRENT` to it. If this operation moves the staging label `AWSCURRENT` from another version to this version, then Secrets Manager also automatically moves the staging label `AWSPREVIOUS` to the version that `AWSCURRENT` was removed from.

This operation is idempotent. If you call this operation with a `ClientRequestToken` that matches an existing version's VersionId, and you specify the same secret data, the operation succeeds but does nothing. However, if the secret data is different, then the operation fails because you can't modify an existing version; you can only create new ones.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters except `SecretBinary`, `SecretString`, or `RotationToken` because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:PutSecretValue`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

> **Important**
>
> When you enter commands in a command shell, there is a risk of the command history being accessed or utilities having access to your command parameters. This is a concern if the command includes the value of a secret. Learn how to Mitigate the risks of using command-line tools to store Amazon Secrets Manager secrets.

Request Syntax

```
{
   "ClientRequestToken": "string",
   "RotationToken": "string",
   "SecretBinary": blob,
   "SecretId": "string",
   "SecretString": "string",
   "VersionStages": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ClientRequestToken

A unique identifier for the new version of the secret.

> **Note**
>
> If you use the Amazon CLI or one of the Amazon SDKs to call this operation, then you can leave this parameter empty. The CLI or SDK generates a random UUID for you and includes it as the value for this parameter in the request.

If you generate a raw HTTP request to the Secrets Manager service endpoint, then you must generate a `ClientRequestToken` and include it in the request.

This value helps ensure idempotency. Secrets Manager uses this value to prevent the accidental creation of duplicate versions if there are failures and retries during a rotation. We recommend that you generate a UUID-type value to ensure uniqueness of your versions within the specified secret.

- If the `ClientRequestToken` value isn't already associated with a version of the secret then a new version of the secret is created.
- If a version with this value already exists and that version's `SecretString` or `SecretBinary` values are the same as those in the request then the request is ignored. The operation is idempotent.

- If a version with this value already exists and the version of the `SecretString` and `SecretBinary` values are different from those in the request, then the request fails because you can't modify a secret version. You can only create new versions to store new secret values.

This value becomes the `VersionId` of the new version.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

RotationToken

A unique identifier that indicates the source of the request. Required for secret rotations using an IAM assumed role or cross-account rotation, in which you rotate a secret in one account by using a Lambda rotation function in another account. In both cases, the rotation function assumes an IAM role to call Secrets Manager, and then Secrets Manager validates the identity using the token. For more information, see How rotation works and Rotation by Lambda functions.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: String

Length Constraints: Minimum length of 36. Maximum length of 256.

Pattern: `^[a-zA-Z0-9\-]+$`

Required: No

SecretBinary

The binary data to encrypt and store in the new version of the secret. To use this parameter in the command-line tools, we recommend that you store your binary data in a file and then pass the contents of the file as a parameter.

You must include `SecretBinary` or `SecretString`, but not both.

You can't access this value from the Secrets Manager console.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

SecretId

The ARN or name of the secret to add a new version to.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

If the secret doesn't already exist, use `CreateSecret` instead.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

SecretString

The text to encrypt and store in the new version of the secret.

You must include `SecretBinary` or `SecretString`, but not both.

We recommend you create the secret string as JSON key/value pairs, as shown in the example.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

VersionStages

A list of staging labels to attach to this version of the secret. Secrets Manager uses staging labels to track versions of a secret through the rotation process.

If you specify a staging label that's already associated with a different version of the same secret, then Secrets Manager removes the label from the other version and attaches it to this version. If you specify `AWSCURRENT`, and it is already attached to another version, then Secrets Manager also moves the staging label `AWSPREVIOUS` to the version that `AWSCURRENT` was removed from.

If you don't include `VersionStages`, then Secrets Manager automatically moves the staging label `AWSCURRENT` to this version.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "VersionId": "string",
   "VersionStages": [ "string" ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

VersionId

The unique identifier of the version of the secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

VersionStages

The list of staging labels that are currently attached to this version of the secret. Secrets Manager uses staging labels to track a version as it progresses through the secret rotation process.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

EncryptionFailure

Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the KMS key is available, enabled, and not in an invalid state. For more information, see Key state: Effect on your KMS key.

HTTP Status Code: 400

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

LimitExceededException

The request failed because it would exceed one of the Secrets Manager quotas.

HTTP Status Code: 400

ResourceExistsException

A resource with the ID you requested already exists.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to create a new version of a secret. The `ClientRequestToken` becomes the `VersionId` of the new version. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.PutSecretValue
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "SecretString": "{\"username\":\"david\",\"password\":\"EXAMPLE-PASSWORD\"}",
  "ClientRequestToken": "EXAMPLE2-90ab-cdef-fedc-ba987EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret",
  "VersionId":"EXAMPLE2-90ab-cdef-fedc-ba987EXAMPLE",
```

```
  "VersionStages":[
      "AWSCURRENT"
  ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

RemoveRegionsFromReplication

For a secret that is replicated to other Regions, deletes the secret replicas from the Regions you specify.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:RemoveRegionsFromReplication`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "RemoveReplicaRegions": [ "string" ],
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

RemoveReplicaRegions

The Regions of the replicas to remove.

Type: Array of strings

Array Members: Minimum number of 1 item.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([a-z]+-)+\d+$`

Required: Yes

SecretId

The ARN or name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "ReplicationStatus": [
      {
         "KmsKeyId": "string",
         "LastAccessedDate": number,
         "Region": "string",
         "Status": "string",
         "StatusMessage": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the primary secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

ReplicationStatus

The status of replicas for this secret after you remove Regions.

Type: Array of ReplicationStatusType objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to remove the replica secrets in Europe (London) and Europe (Paris) The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RemoveRegionsFromReplication
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "RemoveReplicaRegions": "eu-west-2 eu-west-3"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "ReplicationStatus":[
      {
          "Region": "eu-west-1",
          "KmsKeyId": "alias/aws/secretsmanager",
          "Status": "InSync",
          "StatusMessage": "Replication succeeded"

      }
  ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

ReplicateSecretToRegions

Replicates the secret to a new Regions. See Multi-Region secrets.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:ReplicateSecretToRegions`. If the primary secret is encrypted with a KMS key other than `aws/secretsmanager`, you also need `kms:Decrypt` permission to the key. To encrypt the replicated secret with a KMS key other than `aws/secretsmanager`, you need `kms:GenerateDataKey` and `kms:Encrypt` to the key. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "AddReplicaRegions": [
      {
         "KmsKeyId": "string",
         "Region": "string"
      }
   ],
   "ForceOverwriteReplicaSecret": boolean,
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

AddReplicaRegions

A list of Regions in which to replicate the secret.

Type: Array of ReplicaRegionType objects

Array Members: Minimum number of 1 item.

Required: Yes

ForceOverwriteReplicaSecret

Specifies whether to overwrite a secret with the same name in the destination Region. By default, secrets aren't overwritten.

Type: Boolean

Required: No

SecretId

The ARN or name of the secret to replicate.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "ReplicationStatus": [
      {
         "KmsKeyId": "string",
         "LastAccessedDate": number,
         "Region": "string",
         "Status": "string",
         "StatusMessage": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the primary secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

ReplicationStatus

The status of replication.

Type: Array of ReplicationStatusType objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example replicates a secret to eu-west-3. The replica is encrypted with the Amazon managed key `aws/secretsmanager`. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.ReplicateSecretToRegions
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestSecret",
  "AddReplicaRegions": [ { "Region": "eu-west-3" }] ,
  "ForceOverwriteReplicaSecret": true
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
```

```
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestSecret-1a2b3c",
  "ReplicationStatus": [
        {
            "Region": "eu-west-3",
            "KmsKeyId": "alias/aws/secretsmanager",
            "Status": "InProgress"
        }
    ]
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

RestoreSecret

Cancels the scheduled deletion of a secret by removing the `DeletedDate` time stamp. You can access a secret again after it has been restored.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:RestoreSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret to restore.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
```

```
    "ARN": "string",
    "Name": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret that was restored.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret that was restored.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to restore a secret that was previously scheduled for deletion. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RestoreSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

RotateSecret

Configures and starts the asynchronous process of rotating the secret. For information about rotation, see Rotate secrets in the *Secrets Manager User Guide*. If you include the configuration parameters, the operation sets the values for the secret and then immediately starts a rotation. If you don't include the configuration parameters, the operation starts a rotation with the values already stored in the secret.

When rotation is successful, the `AWSPENDING` staging label might be attached to the same version as the `AWSCURRENT` version, or it might not be attached to any version. If the `AWSPENDING` staging label is present but not attached to the same version as `AWSCURRENT`, then any later invocation of `RotateSecret` assumes that a previous rotation request is still in progress and returns an error. When rotation is unsuccessful, the `AWSPENDING` staging label might be attached to an empty secret version. For more information, see Troubleshoot rotation in the *Secrets Manager User Guide*.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:RotateSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager. You also need `lambda:InvokeFunction` permissions on the rotation function. For more information, see Permissions for rotation.

Request Syntax

```
{
   "ClientRequestToken": "string",
   "RotateImmediately": boolean,
   "RotationLambdaARN": "string",
   "RotationRules": {
      "AutomaticallyAfterDays": number,
      "Duration": "string",
      "ScheduleExpression": "string"
   },
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ClientRequestToken

A unique identifier for the new version of the secret. You only need to specify this value if you implement your own retry logic and you want to ensure that Secrets Manager doesn't attempt to create a secret version twice.

> **Note**
>
> If you use the Amazon CLI or one of the Amazon SDKs to call this operation, then you can leave this parameter empty. The CLI or SDK generates a random UUID for you and includes it as the value for this parameter in the request.

If you generate a raw HTTP request to the Secrets Manager service endpoint, then you must generate a `ClientRequestToken` and include it in the request.

This value helps ensure idempotency. Secrets Manager uses this value to prevent the accidental creation of duplicate versions if there are failures and retries during a rotation. We recommend that you generate a UUID-type value to ensure uniqueness of your versions within the specified secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

RotateImmediately

Specifies whether to rotate the secret immediately or wait until the next scheduled rotation window. The rotation schedule is defined in RotateSecret:RotationRules.

The default for `RotateImmediately` is `true`. If you don't specify this value, Secrets Manager rotates the secret immediately.

If you set `RotateImmediately` to `false`, Secrets Manager tests the rotation configuration by running the `testSecret` step of the Lambda rotation function. This test creates an `AWSPENDING` version of the secret and then removes it.

When changing an existing rotation schedule and setting `RotateImmediately` to `false`:

- If using `AutomaticallyAfterDays` or a `ScheduleExpression` with `rate()`, the previously scheduled rotation might still occur.
- To prevent unintended rotations, use a `ScheduleExpression` with `cron()` for granular control over rotation windows.

Rotation is an asynchronous process. For more information, see How rotation works.

Type: Boolean

Required: No

RotationLambdaARN

For secrets that use a Lambda rotation function to rotate, the ARN of the Lambda rotation function.

For secrets that use *managed rotation*, omit this field. For more information, see Managed rotation in the *Secrets Manager User Guide*.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

RotationRules

A structure that defines the rotation configuration for this secret.

> **Important**
>
> When changing an existing rotation schedule and setting `RotateImmediately` to `false`:
>
> - If using `AutomaticallyAfterDays` or a `ScheduleExpression` with `rate()`, the previously scheduled rotation might still occur.
> - To prevent unintended rotations, use a `ScheduleExpression` with `cron()` for granular control over rotation windows.

Type: RotationRulesType object

Required: No

SecretId

The ARN or name of the secret to rotate.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "VersionId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

VersionId

The ID of the new version of the secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example configures rotation for a secret using a cron expression. The first rotation happens immediately after the changes are stored in the secret. The rotation schedule is the first and 15th day of every month. The rotation window begins at 4:00 PM UTC and ends at 6:00 PM. The `ClientRequestToken` field becomes the `VersionId` of the new version created during the rotation. The rotation function runs asynchronously in the background.

The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RotateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestDatabaseRotationLambda",
  "RotationRules": {"ScheduleExpression": "cron(0 16 1,15 * ? *)", "Duration": "2h"},
  "RotateImmediately": true,
  "ClientRequestToken": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
```

```
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Example

The following example shows how to change a rotation schedule safely when setting `RotateImmediately` to `false` and using `cron()` for precise control over rotation timing. This example schedules rotation for the 1st day of each month at 4:00 PM UTC. Setting `RotateImmediately` to `false` prevents an immediate rotation, while using `cron()` prevents unintended rotations that might occur with `rate()` expressions when changing schedules. The `ClientRequestToken` field becomes the `VersionId` of the new version created during the rotation. The rotation function runs asynchronously in the background.

The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RotateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestDatabaseRotationLambda",
```

```
  "RotationRules": {"ScheduleExpression": "cron(0 16 1 * ? *)"},
  "RotateImmediately": false,
  "ClientRequestToken": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Example

The following example configures rotation for a secret using a rate expression. The first rotation happens immediately after the changes are stored in the secret. The rotation schedule is every 10 days. The `ClientRequestToken` field becomes the `VersionId` of the new version created during the rotation. The rotation function runs asynchronously in the background.

The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RotateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
```

```
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "RotationLambdaARN": "arn:aws:lambda:us-
west-2:123456789012:function:MyTestDatabaseRotationLambda",
  "RotationRules": {"ScheduleExpression": "rate(10 days)"},
  "RotateImmediately": true,
  "ClientRequestToken": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Example

The following example starts an immediate rotation, so the secret must already have rotation configured. The `ClientRequestToken` field becomes the `VersionId` of the new version created during the rotation. The rotation function runs asynchronously in the background.

The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.RotateSecret
```

```
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "ClientRequestToken": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN": "arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name": "MyTestDatabaseSecret",
  "VersionId": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin

- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

StopReplicationToReplica

Removes the link between the replica secret and the primary secret and promotes the replica to a primary secret in the replica Region.

You must call this operation from the Region in which you want to promote the replica to a primary secret.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:StopReplicationToReplica`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The name of the secret or the replica ARN. The replica ARN is the same as the original primary secret ARN expect the Region is changed to the replica Region.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Response Syntax

```
{
   "ARN": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the promoted secret. The ARN is the same as the original primary secret except the Region is changed.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.

- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example is intended for a primary secret in `us-west-2` that has a replica in `ap-south-1`. The example removes the replica from the primary secret and promotes it to a primary secret in `ap-south-1`.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.StopReplicationToReplica
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

        {
        "SecretId": "arn:aws:secretsmanager:us-
west-2:123456789012:secret:MyExampleSecret-a1b2c3"
        }
```

Sample Response

```
HTTP/1.1 200 OK
```

```
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

       {
{
  "ARN":"arn:aws:secretsmanager:ap-south-1:123456789012:secret:MyExampleSecret-a1b2c3",
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

TagResource

Attaches tags to a secret. Tags consist of a key name and a value. Tags are part of the secret's metadata. They are not associated with specific versions of the secret. This operation appends tags to the existing list of tags.

For tag quotas and naming restrictions, see Service quotas for Tagging in the *Amazon General Reference guide*.

> **Important**
>
> If you use tags as part of your security strategy, then adding or removing a tag can change permissions. If successfully completing this operation would result in you losing your permissions for this secret, then the operation is blocked and returns an Access Denied error.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:TagResource`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The identifier for the secret to attach tags to. You can specify either the Amazon Resource Name (ARN) or the friendly name of the secret.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

Tags

The tags to attach to the secret as a JSON text string argument. Each element in the list consists of a `Key` and a `Value`.

For storing multiple values, we recommend that you use a JSON text string argument and specify key/value pairs. For more information, see Specifying parameter values for the Amazon CLI in the Amazon CLI User Guide.

Type: Array of Tag objects

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to attach two tags to a secret. There is no output from this API. To see the result, use the DescribeSecret operation.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.TagResource
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
```

```
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyExampleSecret",
  "Tags": [
    {
      "Key": "FirstTag",
      "Value": "SomeValue"
    },
    {
      "Key": "SecondTag",
      "Value": "AnotherValue"
    }
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin

- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

UntagResource

Removes specific tags from a secret.

This operation is idempotent. If a requested tag is not attached to the secret, no error is returned and the secret metadata is unchanged.

> **Important**
>
> If you use tags as part of your security strategy, then removing a tag can change permissions. If successfully completing this operation would result in you losing your permissions for this secret, then the operation is blocked and returns an Access Denied error.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:UntagResource`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "SecretId": "string",
   "TagKeys": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

SecretId

The ARN or name of the secret.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

TagKeys

A list of tag key names to remove from the secret. You don't specify the value. Both the key and its associated value are removed.

This parameter requires a JSON text string argument.

For storing multiple values, we recommend that you use a JSON text string argument and specify key/value pairs. For more information, see Specifying parameter values for the Amazon CLI in the Amazon CLI User Guide.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to remove two tags from secret metadata. For each, both the tag and the associated value are removed. There is no output from this API. To see the result, use the DescribeSecret operation.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UntagResource
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
```

```
  "SecretId": "MyTestDatabaseSecret",
  "TagKeys": [
    "FirstTag", "SecondTag"
  ]
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

UpdateSecret

Modifies the details of a secret, including metadata and the secret value. To change the secret value, you can also use PutSecretValue.

To change the rotation configuration of a secret, use RotateSecret instead.

To change a secret so that it is managed by another service, you need to recreate the secret in that service. See Secrets Manager secrets managed by other Amazon services.

We recommend you avoid calling `UpdateSecret` at a sustained rate of more than once every 10 minutes. When you call `UpdateSecret` to update the secret value, Secrets Manager creates a new version of the secret. Secrets Manager removes outdated versions when there are more than 100, but it does not remove versions created less than 24 hours ago. If you update the secret value more than once every 10 minutes, you create more versions than Secrets Manager removes, and you will reach the quota for secret versions.

If you include `SecretString` or `SecretBinary` to create a new secret version, Secrets Manager automatically moves the staging label `AWSCURRENT` to the new version. Then it attaches the label `AWSPREVIOUS` to the version that `AWSCURRENT` was removed from.

If you call this operation with a `ClientRequestToken` that matches an existing version's `VersionId`, the operation results in an error. You can't modify an existing version, you can only create a new version. To remove a version, remove all staging labels from it. See UpdateSecretVersionStage.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters except `SecretBinary` or `SecretString` because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:UpdateSecret`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager. If you use a customer managed key, you must also have `kms:GenerateDataKey`, `kms:Encrypt`, and `kms:Decrypt` permissions on the key. If you change the KMS key and you don't have `kms:Encrypt` permission to the new key, Secrets Manager does not re-encrypt existing secret versions with the new key. For more information, see Secret encryption and decryption.

Important

When you enter commands in a command shell, there is a risk of the command history being accessed or utilities having access to your command parameters. This is a concern if the command includes the value of a secret. Learn how to Mitigate the risks of using command-line tools to store Amazon Secrets Manager secrets.

Request Syntax

```
{
   "ClientRequestToken": "string",
   "Description": "string",
   "KmsKeyId": "string",
   "SecretBinary": blob,
   "SecretId": "string",
   "SecretString": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ClientRequestToken

If you include `SecretString` or `SecretBinary`, then Secrets Manager creates a new version for the secret, and this parameter specifies the unique identifier for the new version.

Note

If you use the Amazon CLI or one of the Amazon SDKs to call this operation, then you can leave this parameter empty. The CLI or SDK generates a random UUID for you and includes it as the value for this parameter in the request.

If you generate a raw HTTP request to the Secrets Manager service endpoint, then you must generate a `ClientRequestToken` and include it in the request.

This value helps ensure idempotency. Secrets Manager uses this value to prevent the accidental creation of duplicate versions if there are failures and retries during a rotation. We recommend that you generate a UUID-type value to ensure uniqueness of your versions within the specified secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

Description

The description of the secret.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

KmsKeyId

The ARN, key ID, or alias of the KMS key that Secrets Manager uses to encrypt new secret versions as well as any existing versions with the staging labels `AWSCURRENT`, `AWSPENDING`, or `AWSPREVIOUS`. If you don't have `kms:Encrypt` permission to the new key, Secrets Manager does not re-encrypt existing secret versions with the new key. For more information about versions and staging labels, see Concepts: Version.

A key alias is always prefixed by `alias/`, for example `alias/aws/secretsmanager`. For more information, see About aliases.

If you set this to an empty string, Secrets Manager uses the Amazon managed key `aws/secretsmanager`. If this key doesn't already exist in your account, then Secrets Manager creates it for you automatically. All users and roles in the Amazon account automatically have access to use `aws/secretsmanager`. Creating `aws/secretsmanager` can result in a one-time significant delay in returning the result.

> **Important**
>
> You can only use the Amazon managed key `aws/secretsmanager` if you call this operation using credentials from the same Amazon account that owns the secret. If the

secret is in a different account, then you must use a customer managed key and provide the ARN of that KMS key in this field. The user making the call must have permissions to both the secret and the KMS key in their respective accounts.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

SecretBinary

The binary data to encrypt and store in the new version of the secret. We recommend that you store your binary data in a file and then pass the contents of the file as a parameter.

Either `SecretBinary` or `SecretString` must have a value, but not both.

You can't access this parameter in the Secrets Manager console.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

SecretId

The ARN or name of the secret.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

SecretString

The text data to encrypt and store in the new version of the secret. We recommend you use a JSON structure of key/value pairs for your secret value.

Either `SecretBinary` or `SecretString` must have a value, but not both.

Sensitive: This field contains sensitive information, so the service does not include it in Amazon CloudTrail log entries. If you create your own log entries, you must also avoid logging the information in this field.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

Response Syntax

```
{
   "ARN": "string",
   "Name": "string",
   "VersionId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret that was updated.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret that was updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

VersionId

If Secrets Manager created a new version of the secret during this operation, then `VersionId` contains the unique identifier of the new version.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Errors

For information about the errors that are common to all actions, see Common Errors.

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

EncryptionFailure

Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the KMS key is available, enabled, and not in an invalid state. For more information, see Key state: Effect on your KMS key.

HTTP Status Code: 400

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

LimitExceededException

The request failed because it would exceed one of the Secrets Manager quotas.

HTTP Status Code: 400

MalformedPolicyDocumentException

The resource policy has syntax errors.

HTTP Status Code: 400

PreconditionNotMetException

The request failed because you did not complete all the prerequisite steps.

HTTP Status Code: 400

ResourceExistsException

A resource with the ID you requested already exists.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to change the description of a secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "Description": "This is a new description for the secret.",
  "ClientRequestToken": "EXAMPLE1-90ab-cdef-fedc-ba987EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

Example

This example shows how to update the KMS key that Secrets Manager uses to encrypt the secret value. The KMS key must be in the same Region as the secret. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "KmsKeyId": "arn:aws:kms:us-west-2:123456789012:key/EXAMPLE2-90ab-cdef-fedc-
ba987EXAMPLE"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

Example

The following example shows how to create a new version of the secret by updating the `SecretString` field. The `ClientRequestToken` parameter becomes the `VersionId` of the new version. Alternatively, you can use the PutSecretValue operation. The JSON request string input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecret
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "SecretString": "{<JSON STRING WITH CREDENTIALS>}",
  "ClientRequestToken": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret",
  "VersionId":"EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

UpdateSecretVersionStage

Modifies the staging labels attached to a version of a secret. Secrets Manager uses staging labels to track a version as it progresses through the secret rotation process. Each staging label can be attached to only one version at a time. To add a staging label to a version when it is already attached to another version, Secrets Manager first removes it from the other version first and then attaches it to this one. For more information about versions and staging labels, see Concepts: Version.

The staging labels that you specify in the `VersionStage` parameter are added to the existing list of staging labels for the version.

You can move the `AWSCURRENT` staging label to this version by including it in this call.

> **Note**
>
> Whenever you move `AWSCURRENT`, Secrets Manager automatically moves the label `AWSPREVIOUS` to the version that `AWSCURRENT` was removed from.

If this action results in the last label being removed from a version, then the version is considered to be 'deprecated' and can be deleted by Secrets Manager.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:UpdateSecretVersionStage`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "MoveToVersionId": "string",
   "RemoveFromVersionId": "string",
   "SecretId": "string",
   "VersionStage": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

MoveToVersionId

The ID of the version to add the staging label to. To remove a label from a version, then do not specify this parameter.

If the staging label is already attached to a different version of the secret, then you must also specify the `RemoveFromVersionId` parameter.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

RemoveFromVersionId

The ID of the version that the staging label is to be removed from. If the staging label you are trying to attach to one version is already attached to a different version, then you must include this parameter and specify the version that the label is to be removed from. If the label is attached and you either do not specify this parameter, or the version ID does not match, then the operation fails.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

SecretId

The ARN or the name of the secret with the version and staging labelsto modify.

For an ARN, we recommend that you specify a complete ARN rather than a partial ARN. See Finding a secret from a partial ARN.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: Yes

VersionStage

The staging label to add to this version.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

Response Syntax

```
{
   "ARN": "string",
   "Name": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

ARN

The ARN of the secret that was updated.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Name

The name of the secret that was updated.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

LimitExceededException

The request failed because it would exceed one of the Secrets Manager quotas.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to add a staging label to a version of a secret. You can review the results by calling ListSecretVersionIds. The JSON request string input and response output

displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecretVersionStage
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "VersionStage": "STAGINGLABEL1",
  "MoveToVersionId": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

Example

The following example shows you how to remove a staging label from a version of a secret. You can review the results by calling ListSecretVersionIds. The JSON request string input and response

output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecretVersionStage
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "VersionStage": "STAGINGLABEL1",
  "RemoveFromVersionId": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

Example

The following example shows you how to move a staging label from one version of a secret to another. You can review the results by calling ListSecretVersionIds. The JSON request string

input and response output displays formatted code with white space and line breaks for better readability. Submit your input as a single line JSON string.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.UpdateSecretVersionStage
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

{
  "SecretId": "MyTestDatabaseSecret",
  "VersionStage": "AWSCURRENT",
  "RemoveFromVersionId": "EXAMPLE1-90ab-cdef-fedc-ba987SECRET1",
  "MoveToVersionId": "EXAMPLE2-90ab-cdef-fedc-ba987SECRET2"
}
```

Sample Response

```
HTTP/1.1 200 OK
Date: <date>
Content-Type: application/x-amz-json-1.1
Content-Length: <response-size-bytes>
Connection: keep-alive
x-amzn-RequestId: <request-id-guid>

{
  "ARN":"arn:aws:secretsmanager:us-west-2:123456789012:secret:MyTestDatabaseSecret-
a1b2c3",
  "Name":"MyTestDatabaseSecret"
}
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3
- Amazon SDK for Python
- Amazon SDK for Ruby V3

ValidateResourcePolicy

Validates that a resource policy does not grant a wide range of principals access to your secret. A resource-based policy is optional for secrets.

The API performs three checks when validating the policy:

- Sends a call to Zelkova, an automated reasoning engine, to ensure your resource policy does not allow broad access to your secret, for example policies that use a wildcard for the principal.
- Checks for correct syntax in a policy.
- Verifies the policy does not lock out a caller.

Secrets Manager generates a CloudTrail log entry when you call this action. Do not include sensitive information in request parameters because it might be logged. For more information, see Logging Secrets Manager events with Amazon CloudTrail.

Required permissions: `secretsmanager:ValidateResourcePolicy` and `secretsmanager:PutResourcePolicy`. For more information, see IAM policy actions for Secrets Manager and Authentication and access control in Secrets Manager.

Request Syntax

```
{
   "ResourcePolicy": "string",
   "SecretId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters.

The request accepts the following data in JSON format.

ResourcePolicy

A JSON-formatted string that contains an Amazon resource-based policy. The policy in the string identifies who can access or manage this secret and its versions. For example policies, see Permissions policy examples.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 20480.

Required: Yes

SecretId

The ARN or name of the secret with the resource-based policy you want to validate.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

Response Syntax

```
{
   "PolicyValidationPassed": boolean,
   "ValidationErrors": [
      {
         "CheckName": "string",
         "ErrorMessage": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

PolicyValidationPassed

True if your policy passes validation, otherwise false.

Type: Boolean

ValidationErrors

Validation errors if your policy didn't pass validation.

Type: Array of ValidationErrorsEntry objects

Errors

For information about the errors that are common to all actions, see Common Errors.

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 500

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

MalformedPolicyDocumentException

The resource policy has syntax errors.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

Examples

Example

The following example shows how to validate a JSON policy.

Sample Request

```
POST / HTTP/1.1
Host: secretsmanager.region.domain
Accept-Encoding: identity
X-Amz-Target: secretsmanager.ValidateResourcePolicy
Content-Type: application/x-amz-json-1.1
User-Agent: <user-agent-string>
X-Amz-Date: <date>
Authorization: AWS4-HMAC-SHA256 Credential=<credentials>,SignedHeaders=<headers>,
 Signature=<signature>
Content-Length: <payload-size-bytes>

    {
      "SecretId": "MyTestDatabaseSecret",
      "ResourcePolicy": "{\n\"Version\":\"2012-10-17\",\n\"Statement\":[{\n\"Effect\":
\"Allow\",\n\"Principal\":{\n\"AWS\":\"arn:aws:iam::123456789012:root\"\n},\n\"Action
\":\"secretsmanager:GetSecretValue\",\n\"Resource\":\"*\"\n}]\n}"
  }
```

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon Command Line Interface
- Amazon SDK for .NET
- Amazon SDK for C++
- Amazon SDK for Go v2
- Amazon SDK for Java V2
- Amazon SDK for JavaScript V3
- Amazon SDK for Kotlin
- Amazon SDK for PHP V3

- Amazon SDK for Python
- Amazon SDK for Ruby V3

Data Types

The Amazon Secrets Manager API contains several data types that various actions use. This section describes each data type in detail.

> **Note**
>
> The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- APIErrorType
- Filter
- ReplicaRegionType
- ReplicationStatusType
- RotationRulesType
- SecretListEntry
- SecretValueEntry
- SecretVersionsListEntry
- Tag
- ValidationErrorsEntry

APIErrorType

The error Secrets Manager encountered while retrieving an individual secret as part of BatchGetSecretValue.

Contents

ErrorCode

The error Secrets Manager encountered while retrieving an individual secret as part of BatchGetSecretValue, for example `ResourceNotFoundException`,`InvalidParameterException`, `InvalidRequestException`, `DecryptionFailure`, or `AccessDeniedException`.

Type: String

Required: No

Message

A message describing the error.

Type: String

Required: No

SecretId

The ARN or name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++

- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

Filter

Allows you to add filters when you use the search function in Secrets Manager. For more information, see Find secrets in Secrets Manager.

Contents

Key

The following are keys you can use:

- **description**: Prefix match, not case-sensitive.
- **name**: Prefix match, case-sensitive.
- **tag-key**: Prefix match, case-sensitive.
- **tag-value**: Prefix match, case-sensitive.
- **primary-region**: Prefix match, case-sensitive.
- **owning-service**: Prefix match, case-sensitive.
- **all**: Breaks the filter value string into words and then searches all attributes for matches. Not case-sensitive.

Type: String

Valid Values: `description | name | tag-key | tag-value | primary-region | owning-service | all`

Required: No

Values

The keyword to filter for.

You can prefix your search value with an exclamation mark (`!`) in order to perform negation filters.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 10 items.

Length Constraints: Maximum length of 512.

Pattern: `^\!?[a-zA-Z0-9 :_@\/\+\=\.\-\!]*$`

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

ReplicaRegionType

A custom type that specifies a `Region` and the `KmsKeyId` for a replica secret.

Contents

KmsKeyId

The ARN, key ID, or alias of the KMS key to encrypt the secret. If you don't include this field, Secrets Manager uses `aws/secretsmanager`.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

Region

A Region code. For a list of Region codes, see Name and code of Regions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([a-z]+-)+\d+$`

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

ReplicationStatusType

A replication object consisting of a `RegionReplicationStatus` object and includes a Region, KMSKeyId, status, and status message.

Contents

KmsKeyId

Can be an `ARN`, `Key ID`, or `Alias`.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

LastAccessedDate

The date that the secret was last accessed in the Region. This field is omitted if the secret has never been retrieved in the Region.

Type: Timestamp

Required: No

Region

The Region where replication occurs.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([a-z]+-)+\d+$`

Required: No

Status

The status can be `InProgress`, `Failed`, or `InSync`.

Type: String

Valid Values: `InSync | Failed | InProgress`

Required: No

StatusMessage

Status message such as "*Secret with this name already exists in this region*".

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4096.

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

RotationRulesType

A structure that defines the rotation configuration for the secret.

Contents

AutomaticallyAfterDays

The number of days between rotations of the secret. You can use this value to check that your secret meets your compliance guidelines for how often secrets must be rotated. If you use this field to set the rotation schedule, Secrets Manager calculates the next rotation date based on the previous rotation. Manually updating the secret value by calling `PutSecretValue` or `UpdateSecret` is considered a valid rotation.

In `DescribeSecret` and `ListSecrets`, this value is calculated from the rotation schedule after every successful rotation. In `RotateSecret`, you can set the rotation schedule in `RotationRules` with `AutomaticallyAfterDays` or `ScheduleExpression`, but not both. To set a rotation schedule in hours, use `ScheduleExpression`.

Type: Long

Valid Range: Minimum value of 1. Maximum value of 1000.

Required: No

Duration

The length of the rotation window in hours, for example `3h` for a three hour window. Secrets Manager rotates your secret at any time during this window. The window must not extend into the next rotation window or the next UTC day. The window starts according to the `ScheduleExpression`. If you don't specify a `Duration`, for a `ScheduleExpression` in hours, the window automatically closes after one hour. For a `ScheduleExpression` in days, the window automatically closes at the end of the UTC day. For more information, including examples, see Schedule expressions in Secrets Manager rotation in the *Secrets Manager Users Guide*.

Type: String

Length Constraints: Minimum length of 2. Maximum length of 3.

Pattern: `[0-9]+h`

Required: No

ScheduleExpression

A `cron()` or `rate()` expression that defines the schedule for rotating your secret. Secrets Manager rotation schedules use UTC time zone. Secrets Manager rotates your secret any time during a rotation window.

Secrets Manager `rate()` expressions represent the interval in hours or days that you want to rotate your secret, for example `rate(12 hours)` or `rate(10 days)`. You can rotate a secret as often as every four hours. If you use a `rate()` expression, the rotation window starts at midnight. For a rate in hours, the default rotation window closes after one hour. For a rate in days, the default rotation window closes at the end of the day. You can set the `Duration` to change the rotation window. The rotation window must not extend into the next UTC day or into the next rotation window.

You can use a `cron()` expression to create a rotation schedule that is more detailed than a rotation interval. For more information, including examples, see Schedule expressions in Secrets Manager rotation in the *Secrets Manager Users Guide*. For a cron expression that represents a schedule in hours, the default rotation window closes after one hour. For a cron expression that represents a schedule in days, the default rotation window closes at the end of the day. You can set the `Duration` to change the rotation window. The rotation window must not extend into the next UTC day or into the next rotation window.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[0-9A-Za-z\(\)#\?\*\-\/, ]+`

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

SecretListEntry

A structure that contains the details about a secret. It does not include the encrypted `SecretString` and `SecretBinary` values. To get those values, use GetSecretValue .

Contents

ARN

The Amazon Resource Name (ARN) of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

CreatedDate

The date and time when a secret was created.

Type: Timestamp

Required: No

DeletedDate

The date and time the deletion of the secret occurred. Not present on active secrets. The secret can be recovered until the number of days in the recovery window has passed, as specified in the `RecoveryWindowInDays` parameter of the `DeleteSecret` operation.

Type: Timestamp

Required: No

Description

The user-provided description of the secret.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

KmsKeyId

The ARN of the Amazon KMS key that Secrets Manager uses to encrypt the secret value. If the secret is encrypted with the Amazon managed key `aws/secretsmanager`, this field is omitted.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

LastAccessedDate

The date that the secret was last accessed in the Region. This field is omitted if the secret has never been retrieved in the Region.

Type: Timestamp

Required: No

LastChangedDate

The last date and time that this secret was modified in any way.

Type: Timestamp

Required: No

LastRotatedDate

The most recent date and time that the Secrets Manager rotation process was successfully completed. This value is null if the secret hasn't ever rotated.

Type: Timestamp

Required: No

Name

The friendly name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

NextRotationDate

The next rotation is scheduled to occur on or before this date. If the secret isn't configured for rotation or rotation has been disabled, Secrets Manager returns null.

Type: Timestamp

Required: No

OwningService

Returns the name of the service that created the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: No

PrimaryRegion

The Region where Secrets Manager originated the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `^([a-z]+-)+\d+$`

Required: No

RotationEnabled

Indicates whether automatic, scheduled rotation is enabled for this secret.

Type: Boolean

Required: No

RotationLambdaARN

The ARN of an Amazon Lambda function invoked by Secrets Manager to rotate and expire the secret either automatically per the schedule or manually by a call to `RotateSecret`.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

RotationRules

A structure that defines the rotation configuration for the secret.

Type: RotationRulesType object

Required: No

SecretVersionsToStages

A list of all of the currently assigned `SecretVersionStage` staging labels and the `SecretVersionId` attached to each one. Staging labels are used to keep track of the different versions during the rotation process.

> **Note**
>
> A version that does not have any `SecretVersionStage` is considered deprecated and subject to deletion. Such versions are not included in this list.

Type: String to array of strings map

Key Length Constraints: Minimum length of 32. Maximum length of 64.

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

Tags

The list of user-defined tags associated with the secret. To add tags to a secret, use `TagResource`. To remove tags, use `UntagResource`.

Type: Array of Tag objects

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

SecretValueEntry

A structure that contains the secret value and other details for a secret.

Contents

ARN

The Amazon Resource Name (ARN) of the secret.

Type: String

Length Constraints: Minimum length of 20. Maximum length of 2048.

Required: No

CreatedDate

The date the secret was created.

Type: Timestamp

Required: No

Name

The friendly name of the secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

SecretBinary

The decrypted secret value, if the secret value was originally provided as binary data in the form of a byte array. The parameter represents the binary data as a base64-encoded string.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

SecretString

The decrypted secret value, if the secret value was originally provided as a string or through the Secrets Manager console.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 65536.

Required: No

VersionId

The unique version identifier of this version of the secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

VersionStages

A list of all of the staging labels currently attached to this version of the secret.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

SecretVersionsListEntry

A structure that contains information about one version of a secret.

Contents

CreatedDate

The date and time this version of the secret was created.

Type: Timestamp

Required: No

KmsKeyIds

The KMS keys used to encrypt the secret version.

Type: Array of strings

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

LastAccessedDate

The date that this version of the secret was last accessed. Note that the resolution of this field is at the date level and does not include the time.

Type: Timestamp

Required: No

VersionId

The unique version identifier of this version of the secret.

Type: String

Length Constraints: Minimum length of 32. Maximum length of 64.

Required: No

VersionStages

An array of staging labels that are currently associated with this version of the secret.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 20 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

Tag

A structure that contains information about a tag.

Contents

Key

The key identifier, or name, of the tag.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 128.

Required: No

Value

The string value associated with the key of the tag.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

ValidationErrorsEntry

Displays errors that occurred during validation of the resource policy.

Contents

CheckName

Checks the name of the policy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 512.

Required: No

ErrorMessage

Displays error messages if validation encounters problems during validation of the resource policy.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific Amazon SDKs, see the following:

- Amazon SDK for C++
- Amazon SDK for Java V2
- Amazon SDK for Ruby V3

Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing Amazon API requests in the *IAM User Guide*.

Action

The action to be performed.

Type: string

Required: Yes

Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: *access_key*/*YYYYMMDD*/*region*/*service*/aws4_request.

For more information, see Create a signed Amazon API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-Date

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: `20120325T120000Z`.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an Amazon API request signature in the *IAM User Guide*.

Type: string

Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to Amazon Security Token Service (Amazon STS). For a list of services that support temporary security credentials from Amazon STS, see Amazon Web Services services that work with IAM in the *IAM User Guide*.

Condition: If you're using temporary security credentials from Amazon STS, you must include the security token.

Type: string

Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Create a signed Amazon API request in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

Common Errors

This section lists the errors that can occur while using Amazon Secrets Manager.

AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

DecryptionFailure

Secrets Manager can't decrypt the protected secret text using the provided KMS key.

HTTP Status Code: 400

EncryptionFailure

Secrets Manager can't encrypt the protected secret text using the provided KMS key. Check that the KMS key is available, enabled, and not in an invalid state. For more information, see Key state: Effect on your KMS key.

HTTP Status Code: 400

IncompleteSignature

The request signature does not conform to Amazon standards.

HTTP Status Code: 400

InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InternalServiceError

An error occurred on the server side.

HTTP Status Code: 400

InternalServiceFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

InvalidClientTokenId

The X.509 certificate or Amazon access key ID provided does not exist in our records.

HTTP Status Code: 403

InvalidNextTokenException

The `NextToken` value is invalid.

HTTP Status Code: 400

InvalidParameterException

The parameter name or value is invalid.

HTTP Status Code: 400

InvalidRequestException

A parameter value is not valid for the current state of the resource.

Possible causes:

- The secret is scheduled for deletion.
- You tried to enable rotation on a secret that doesn't already have a Lambda function ARN configured and you didn't include such an ARN as a parameter in this call.
- The secret is managed by another service, and you must use that service to update it. For more information, see Secrets managed by other Amazon services.

HTTP Status Code: 400

LimitExceededException

The request failed because it would exceed one of the Secrets Manager quotas.

HTTP Status Code: 400

MalformedPolicyDocumentException

The resource policy has syntax errors.

HTTP Status Code: 400

NotAuthorized

You do not have permission to perform this action.

HTTP Status Code: 400

OptInRequired

The Amazon access key ID needs a subscription for the service.

HTTP Status Code: 403

PreconditionNotMetException

The request failed because you did not complete all the prerequisite steps.

HTTP Status Code: 400

PublicPolicyException

The `BlockPublicPolicy` parameter is set to true, and the resource policy did not prevent broad access to the secret.

HTTP Status Code: 400

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ResourceExistsException

A resource with the ID you requested already exists.

HTTP Status Code: 400

ResourceNotFoundException

Secrets Manager can't find the resource that you asked for.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by the service.

HTTP Status Code: 400

ValidationException

The input fails to satisfy the constraints specified by the service.

HTTP Status Code: 400