AWS Secrets Manager

Help Panel

AWS Secrets Manager: Help Panel

Table of Contents

Choose an AWS Lambda function

Important
If you enable rotation, Secrets Manager immediately rotates the credentials in the secret once to validate the new configuration. Ensure that all of your applications that use these credentials are updated to retrieve the credentials from this secret using Secrets Manager.

To rotate a secret for a non-RDS database or for a custom secret type, you must create and configure an AWS Lambda function that rotates the secrets when triggered. The rotation function updates the credentials on the protected service and updates the secret to match. Your applications then immediately begin accessing the protected service by using the new credentials contained in the secret.

Choose the Lambda function that contains the code that can rotate your secret.

You can view or choose only functions for which both you and the Secrets Manager service (secretsmanager.amazonaws.com) have the `lambda:InvokeFunction`permission. Alternatively, if the function doesn't exist yet, choose **Create function** to go to the AWS Lambda console to create the function. When you return to this window, choose the refresh button to see the new function in the list.

Learn more

- Rotating secrets
- Example rotation functions

Configure automatic rotation

Important
If you enable rotation, Secrets Manager immediately rotates the secret to test the configuration. Ensure that all of your applications that use these credentials are updated to retrieve the credentials from this secret using Secrets Manager. After the initial rotation, Secrets Manager begins rotating the secret according to the schedule you specify.

If your secret contains credentials, you can configure Secrets Manager to automatically rotate those credentials on a schedule that you specify. Rotation helps keep your IT resources and data secure by regularly changing the credentials. This helps to reduce the risk from leaving your credentials unchanged for long periods of time.

To rotate a secret for a non-RDS database or for a custom secret type, you must create and configure an AWS Lambda function that rotates the secrets when triggered. The rotation function updates the credentials on the protected service, and updates the secret to match. Your applications then immediately begin accessing the protected service by using the new credentials contained in the secret.

Learn more

- Rotating secrets
- Example rotation functions

Rotation configuration

You can rotate the credentials manually or modify the automatic rotation schedule.

To rotate the credentials now, choose **Rotate secret immediately**.

To modify the rotation configuration for this secret, choose **Edit rotation**.

You must have the appropriate permissions to configure rotation for a secret.

> **Important**
> If you change the rotation configuration, Secrets Manager immediately rotates the secret value in the secret one time to validate the new settings. Ensure that all of your applications that use the secret value are updated to retrieve the information from this secret using Secrets Manager.

Learn more

Secret name and description

Provide a name for the secret. The name can include a path with / characters to enable you to logically group your secrets. This makes performing some operations easier, such as setting permissions, by enabling you to reference all of the secrets in a path with strings like “/pathname/*”. You must include the path in any reference to the secret, including those made by your users and client apps.

A clear description is always helpful in remembering what a particular secret was intended for long after it was created.

Secrets details

Use this page to view all of the information about your secret. You can also rotate, delete, or edit some of the details of the secret.

You can edit the secret's description, select a new AWS KMS customer master key, or delete the secret by choosing an option on the **Actions** menu.

The secret value that's encrypted when the secret is stored isn't automatically decrypted and displayed. You must choose **Retrieve secret value** to request that Secrets Manager decrypt and display the secret value.

You can also enable and configure automatic rotation for your secret. If you alter your rotation configuration, Secrets Manager immediately rotates your credentials one time to validate the settings.

Learn more

Secrets list

Your secrets are listed on this page. Select a secret to read the details or to configure.

Learn more

Secret value

Choose **Retrieve secret value** to ask Secrets Manager to decrypt your protected credentials and display them in the console. You will have the option to close the section to keep the data hidden until you're ready to view.

Most secrets are stored as JSON key/value pairs. The **Secret key/value** tab shows them interpreted that way. To see the raw, unparsed text, choose the **Plaintext** tab.

Learn more

Select rotation interval

Choose the number of days between rotations of this secret. AWS Secrets Manager automatically triggers a rotation this number of days after the previous rotation. If you ever rotate the secret manually, the rotation interval resets.

You can choose one of the predefined values, or choose **Custom** and then type any number between 1 and the maximum value of 365.

Learn more

- Rotating secrets

Select secret type

Credentials for an RDS database

Provide the user name and password (credentials), and choose the RDS instance that you'll access using these credentials. You also specify which AWS KMS key is used to encrypt the secured information in the secret.

Credentials for a Redshift cluster

Provide the user name and password (credentials) and choose the Redshift cluster that you'll access using these credentials. You also specify which AWS KMS key is used to encrypt the secured information in the secret.

Credentials for a DocumentDB database

Provide the user name and password (credentials) and choose the DocumentDB database that you'll access using these credentials. You also specify which AWS KMS key is used to encrypt the secured information in the secret.

Credentials for other database

Choose this option if you want to store a secret for a database that isn't managed by AWS. When you're storing the secret, you'll provide the user name and password (credentials) for the database that you'd use to access the database. You'll also specify the type of database, the IP address, the database name, and the TCP port number that the database uses to listen for requests. You also specify which AWS KMS key is used to encrypt the secured information in the secret. Secrets Manager requires this information to connect to the database to update the credentials during rotation.

Learn more

Storing a secret for another service

The **Other type of secrets** option enables you to create a secret that can store credentials or other information for any type of service.

You specify the information by defining key-value strings that are stored in the secret.

If you enable rotation, you must create a custom Lambda rotation function that can rotate the secret information that you store in this secret. Your secret should include both the credentials and the connection details that are necessary for your custom Lambda rotation function to access the service and make the updates required by a rotation.

Select the encryption key

Your secret information is encrypted using encryption keys that you can manage by using AWS KMS. You can encrypt by using the default service encryption key that Secrets Manager creates on your behalf. Alternatively, you can encrypt by using a customer master key (CMK) that you create in AWS KMS.

If you use a custom CMK, then the IAM user or role that needs to read the secret later must have the permission “kms:Decrypt” for that KMS CMK.

You’re not billed for using the default encryption key that Secrets Manager creates for you. You’re billed only for your use of CMKs that you create.

Select which database this secret will access

Choose the database product that supports your database.

Type the IPv4 address of the database's host, the name of the database, and the TCP port number on which the selected database listens for requests.

This connection information is stored in the secret with the user name and password. The rotation Lambda function uses the connection information to connect to the database and update the user's credentials.

Select which DocumentDB database this secret will access

The table below lists all the supported DocumentDB databases available in the current AWS account. You can filter the list by typing the first few letters in the search box. Choose the instance that the credentials are intended for. Secrets Manager automatically gets the connection details for the chosen database and stores them in the secret, along with the user name and password. If you choose to enable rotation, Secrets Manager requires this information to connect to the database to update the credentials during rotation.

Learn more.

Select which RDS database this secret will access

Secrets Manager supports native rotation for MySQL, PostgreSQL, Oracle, SQL Server, MariaDB, Aurora MySQL, and Aurora PostgreSQL databases hosted on RDS. The table below lists all the supported RDS databases that are available in the current AWS account. You can filter the list by typing the first few letters in the search box. Choose the instance that the credentials are intended for. Secrets Manager automatically gets the connection details for the chosen database and stores them in the secret, along with the user name and password. If you choose to enable rotation, Secrets Manager requires this information to connect to the database to update the credentials during rotation.

Learn more

Select which Redshift cluster this secret will access

The table below lists all the supported Redshift clusters that are available in the current AWS account. You can filter the list by typing the first few letters in the search box. Choose the instance that the credentials are intended for. Secrets Manager automatically gets the connection details for the chosen database and stores them in the secret, along with the user name and password. If you choose to enable rotation, Secrets Manager requires this information to connect to the database to update the credentials during rotation.

Learn more

Specify the key/value pairs to be stored for this secret

Provide your secret information, such as credentials and connection details, as key name and value string pairs. For example, you could specify “UserName” as a key name and the user’s sign-in name as the value. Another example could be the key name “Password” and the actual password as the value. You can also create pairs for “IpAddress”, “TcpPort”, “ServiceName”, or any other value that’s useful. The Lambda rotation function parses and uses these details to rotate the credentials.

Choose **+Add row** if you need to add more key-value pairs, up to the maximum size limit for a secret.

You can choose the **Plaintext** option to view all of the pairs as the JSON plaintext string that’s stored in the secret.

Store a new secret

AWS Secrets Manager helps you protect access to your IT resources and data by enabling you to easily rotate, manage, and retrieve database credentials, API keys, and other secrets. Start by choosing the type of information you want to protect.

- If you want to protect credentials that are used to access a database hosted on Amazon RDS, choose the **Credentials for RDS database** option.
- If you want to protect credentials that are used to access a database hosted outside of Amazon RDS (for example, on an Amazon EC2 instance that you manage or in an on-premises data center), choose the **Credentials for other database** option.
- If you want to protect any other type of secret, such as an API key or an OAuth token, choose the **Other type of secret** option

You can configure your secret to automatically rotate on a schedule that you specify. Rotation helps keep your IT resources and data secure by regularly changing the secrets. This helps to reduce the risk of leaving a secret unchanged for long periods of time.

Learn more

- Create a basic secret
- Tutorial: Storing and retrieving a secret
- Terms and concepts

User name and password

Enter the user name and password that grant access to the database.

The rules of the associated database determine the maximum length and available characters for the user name and password. We recommend that you always use a password that's as long and complex as your database supports.