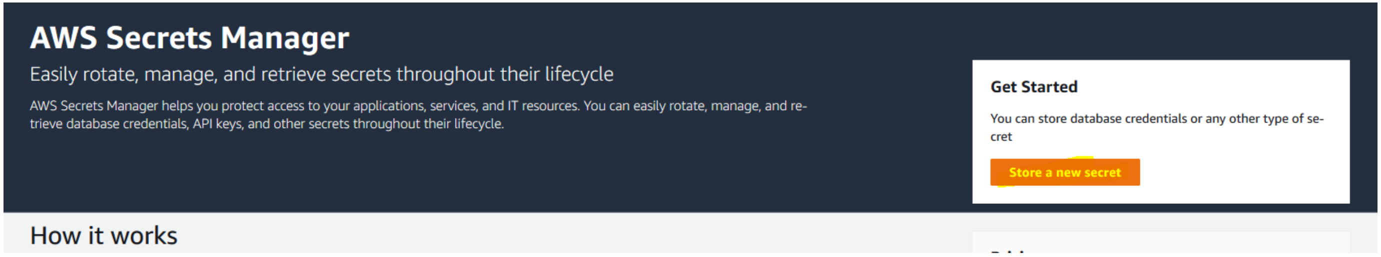
Create and store Secrets

Create and store Secrets.

Step 1) Login to AWS console and open the secrets manager service. Select store a new secret to store the secret.



AWS Secrets Manager

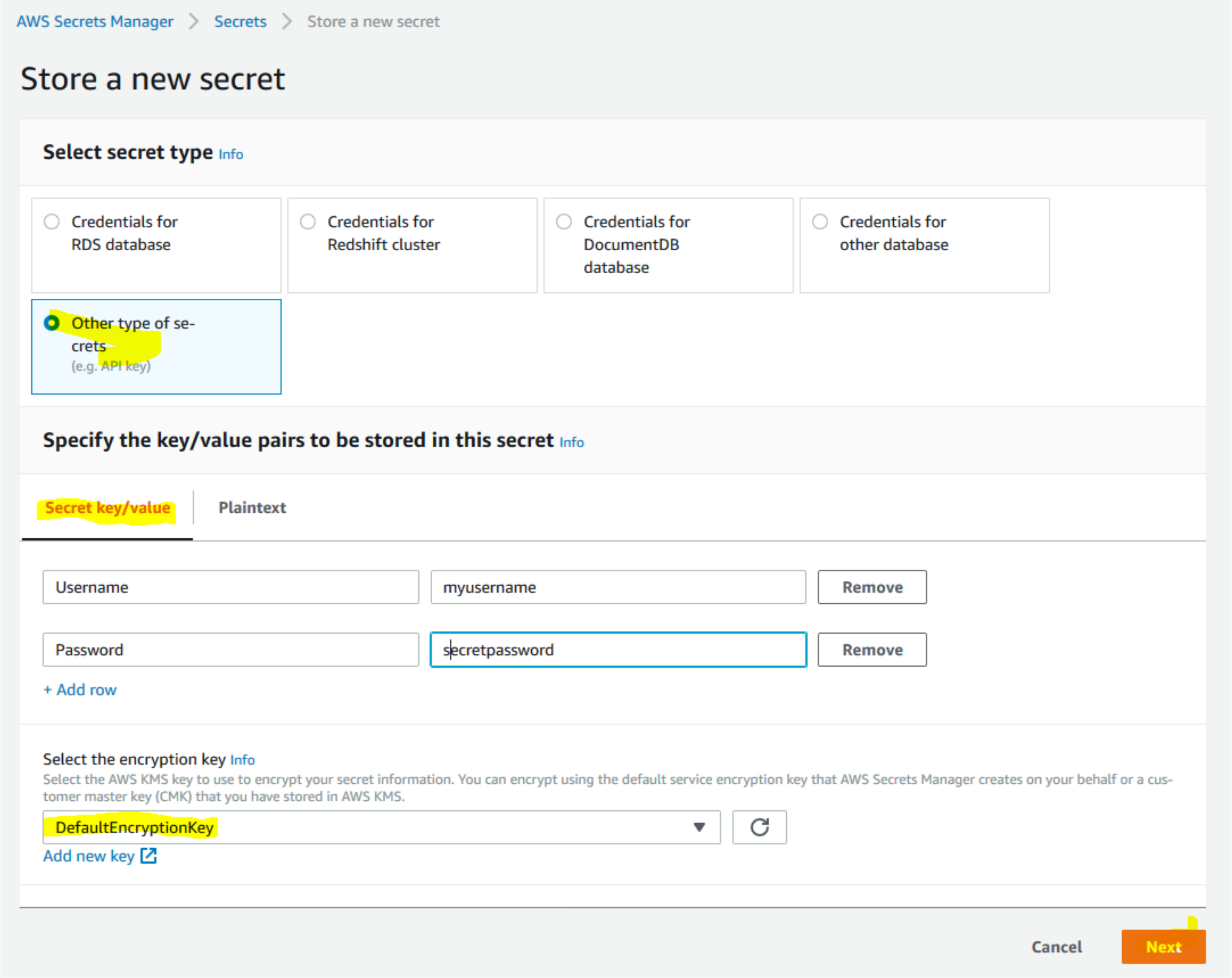
Easily rotate, manage, and retrieve secrets throughout their lifecycle

AWS Secrets Manager helps you protect access to your applications, services, and IT resources. You can easily rotate, manage, and retrieve database credentials, API keys, and other secrets throughout their lifecycle.

Get Started

You can store database credentials or any other type of secret

Step 2) Select the type of secret that needs to be stored. I have selected the other type of secrets for the demo purpose. Then select the key value that needs to be stored and select the type of encryption.



Store a new secret

Select secret type Info

○ Credentials for RDS database

○ Credentials for Redshift cluster

○ Credentials for DocumentDB database

○ Credentials for other database

● Other type of secrets (e.g. API key)

Specify the key/value pairs to be stored in this secret Info

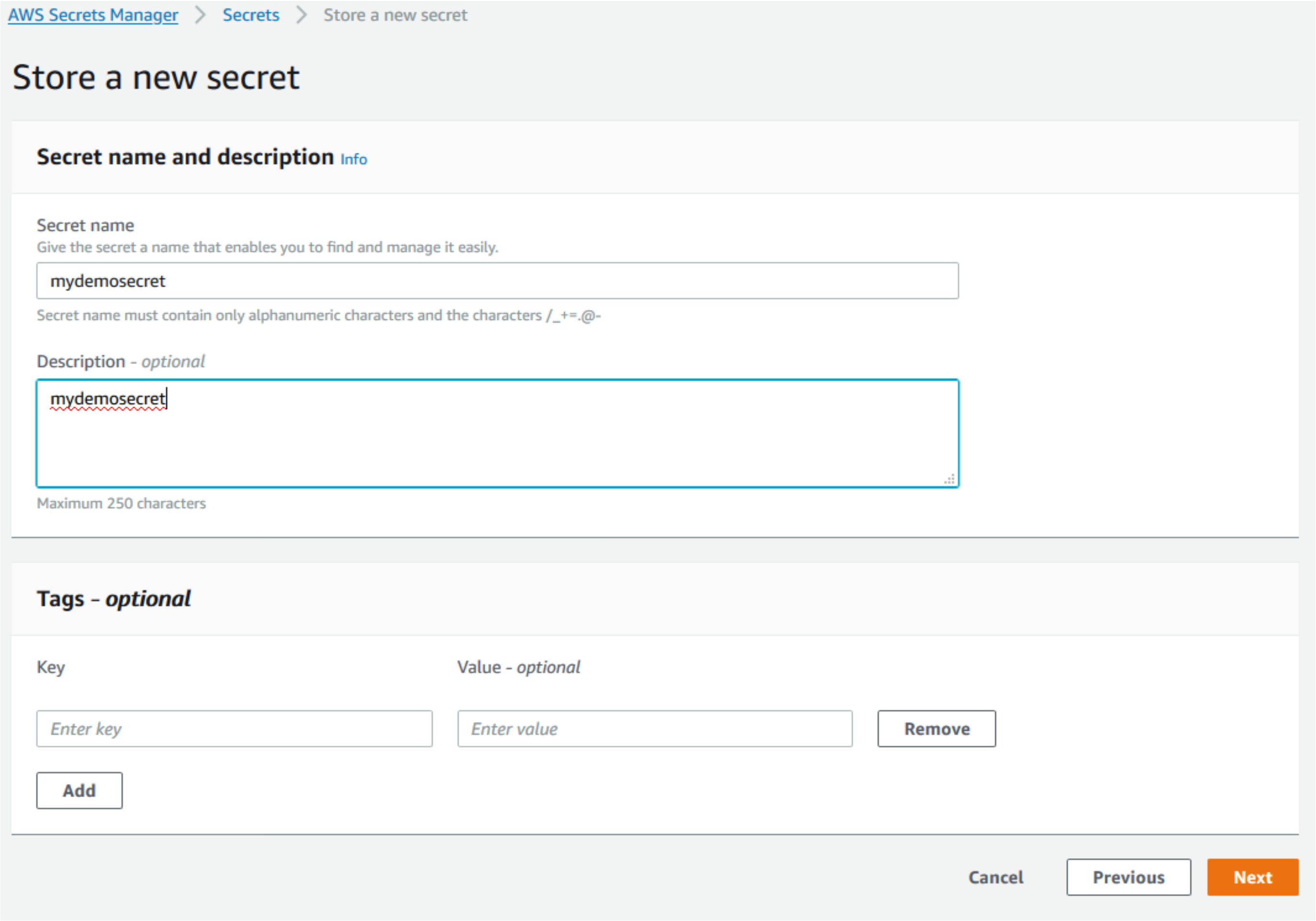
Username	myusername	Remove
Password	secretpassword	Remove

Select the encryption key Info

Select the AWS KMS key to use to encrypt your secret information. You can encrypt using the default service encryption key that AWS Secrets Manager creates on your behalf or a customer master key (CMK) that you have stored in AWS KMS.

DefaultEncryptionKey

Step 3) Provide the secret name and the description.



Store a new secret

Secret name and description Info

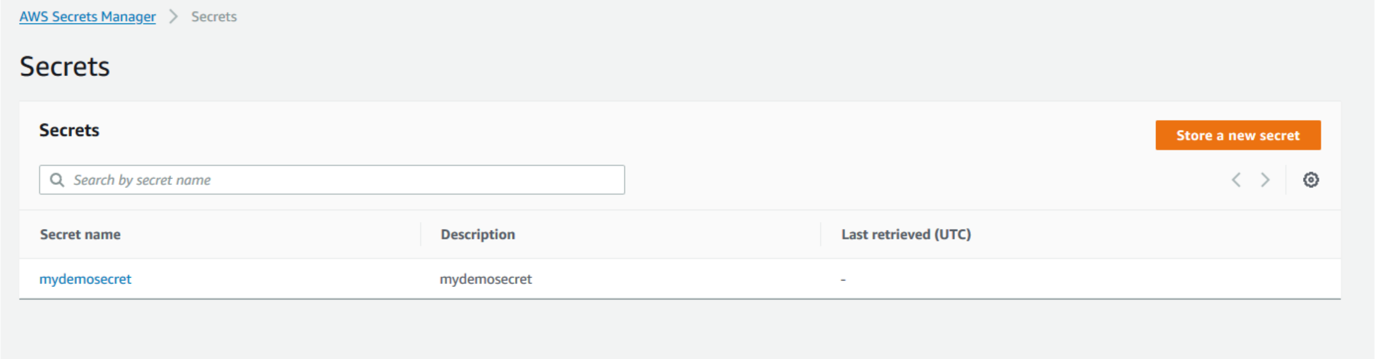
Secret name
Give the secret a name that enables you to find and manage it easily.
mydemosecret
Secret name must contain only alphanumeric characters and the characters /_+=.@-

Description - *optional*
mydemosecret
Maximum 250 characters

Tags - *optional*

Key	Value - *optional*
Enter key	*Enter value*

Step 4) Secret is created and stored.



Secrets

Secrets

Secret name	Description	Last retrieved (UTC)
mydemosecret	mydemosecret	-

Through CLI-

```
[root@ip-10-0-1-238 ~]# aws2 secretsmanager create-secret --name MydummyCLI --description "Secret generated using CLI" --secret-string file://mycreds.json

{

"ARN": "arn:aws:secretsmanager:us-east-1:536285340728:secret:MydummyCLI-cJq19X",

"Name": "MydummyCLI",

"VersionId": "81171c6f-a25d-4c25-97f1-c30043f7b5fc"

}
```

Listing the secrets-

```
[root@ip-10-0-1-238 ~]# aws2 secretsmanager list-secrets {

"SecretList": [
{
"ARN": "arn:aws:secretsmanager:us-east-1:536285340728:secret:mydemosecret-Xv0EYR",
"Name": "mydemosecret",
"Description": "mydemosecret",
"LastChangedDate": "2019-12-04T05:16:09.158000+00:00",
"LastAccessedDate": "2019-12-04T00:00:00+00:00",
"Tags": [],
"SecretVersionsToStages": {
"d89088d5-6be4-4559-911e-2cff81b2a200": [
"AWSCURRENT"
]
}
},
{
"ARN": "arn:aws:secretsmanager:us-east-1:536285340728:secret:demo/secretsmanager-1bKdjz",
"Name": "demo/secretsmanager",
"Description": "demo for password rotation.",
"RotationLambdaARN": "arn:aws:lambda:us-east-1:536285340728:function:SecretsManagerRotateExample",
```

To delete a secret

The following example shows how to delete a secret. The secret stays in your account in a deprecated and inaccessible state until the recovery window ends. After the date and time in the DeletionDate response field has passed, you can no longer recover this secret with restore-secret.

```
aws secretsmanager delete-secret --secret-id MydummyCLI \
--recovery-window-in-days 7
```