

## Tf-adv-secretsmgr

### Steps

1. Create an secrets mgr on aws
2. Create github repo and clone it locally
3. Copy the lab 3 code to this repo folder
4. Configure the secret mgr name in the tf main file

#### 1. Create an secrets mgr on aws

The screenshot shows the 'Choose secret type' step of the AWS Secrets Manager wizard. The left sidebar lists steps: Step 1 (selected), Step 2, Step 3 - optional, and Step 4. The main area shows 'Secret type' options: Credentials for Amazon RDS database, Credentials for Amazon DocumentDB database, Credentials for Amazon Redshift data warehouse, Credentials for other database, Managed external secret, and Other type of secret (selected). A blue arrow points to the 'Other type of secret' option.

The screenshot shows the 'Key/value pairs' step. It has tabs for 'Key/value' (selected) and 'Plaintext'. A row is shown with key 'api\_key' and value 'Password123!'. A blue arrow points to the 'api\_key' field. Below is a ' + Add row' button.

The screenshot shows the 'Encryption key' step. It says you can encrypt using a KMS key. A dropdown shows 'aws/secretsmanager' selected, and a 'Add new key' button is below it. A blue arrow points to the dropdown.

Cancel      Next

## Configure secret

### Secret name and description Info

#### Secret name

A descriptive name that helps you find your secret later.

Secret name must only contain alphanumeric characters and the characters /\_+=@-

#### Description - optional

Maximum 250 characters.

### Tags - optional

No tags associated with the secret.

[Add](#)

### Resource permissions - optional Info

[Edit permissions](#)

Add or edit a resource policy to access secrets across AWS accounts.

### ► Replicate secret - optional

Create read-only replicas of your secret in other regions. Replica secrets incur a charge.

[Cancel](#)[Previous](#)[Next](#)

Assign a name -- **dev/app/api\_key**

### Configure rotation - optional

#### Configure automatic rotation Info

Configure AWS Secrets Manager to rotate this secret automatically.

 Automatic rotation

#### Rotation schedule Info

 Schedule expression builder schedule expression

##### Time unit

##### Hours

##### Window duration - optional

Enter the time in hours.

Rotate immediately when the secret is stored. The next rotation will begin on your schedule.

#### Rotation function Info

##### Lambda rotation function Info

Choose a Lambda function that can rotate this secret.

[Create function](#)[Cancel](#)[Previous](#)[Next](#)

Click next

Secret that performs rotation

-

### Sample code

Use these code samples to retrieve the secret in your application.

[Java](#) | [JavaScript](#) | [C#](#) | [Python3](#) | [Ruby](#) | [Go](#) | [Rust](#) | [PHP](#)

```

1 // Use this code snippet in your app.
2 // If you need more information about configurations or implementing the sample
3 // code, visit the AWS docs:
4 // https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/home.html
5
6 // Make sure to import the following packages in your code
7 // import software.amazon.awssdk.regions.Region;
8 // import software.amazon.awssdk.services.secretsmanager.SecretsManagerClient;
9 // import software.amazon.awssdk.services.secretsmanager.model.GetSecretValueRequest;
10 // import software.amazon.awssdk.services.secretsmanager.model.GetSecretValueResponse;
11
12 public static void getSecret() {
13
14     String secretName = "dev/app/api_key";
15     Region region = Region.of("us-west-2");

```

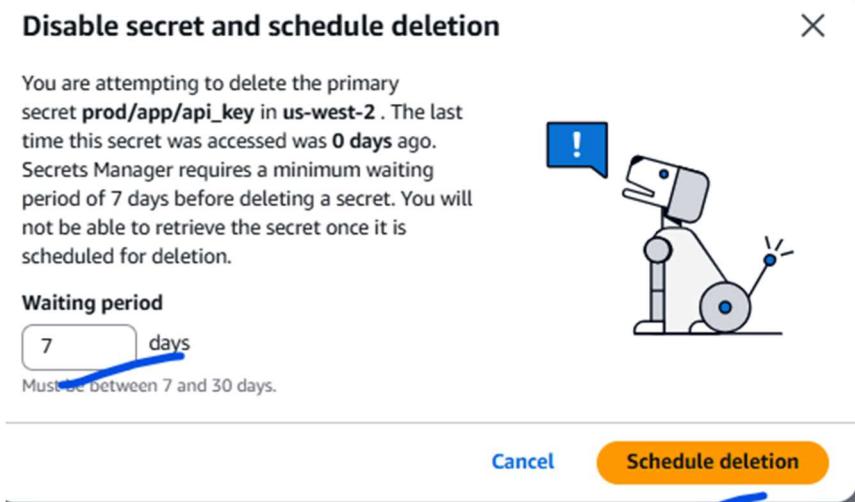
Java Line 1, column 1 | [Errors: 0](#) | [Warnings: 0](#)

[Download AWS SDK for Java](#)

[Cancel](#) [Previous](#) [Store](#)

Click on **store**.

**Note: -- Delete details.**



**Deleting the secrets is min of 7 days**

2. Create github repo and clone it locally

## Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? [Import a repository](#).  
Required fields are marked with an asterisk (\*).

1 General

Owner \* Vishwanathms / Repository name \* tf-adv-secrets 

Great repository names are short and memorable. How about [curly-octo-memory](#)?

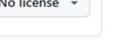
Description tf-adv-secrets   
14 / 350 characters

2 Configuration

Choose visibility \* Public 

Add README Off 

Add .gitignore No .gitignore 

Add license No license 

**Create repository** 

```
● [ec2-user@ip-172-31-55-05 ~]$ cd tf-adv-secrets/
● [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ ls
● [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ ls -a
. .. .git
○ [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ 
```

Cloned on the vs code.

3. Copy the lab 3 code to this repo folder

```

        > Lab2
        >   > Lab3
        >     > infra
        >       > backend.tf
        >       > main.tf
        >       > userdata.sh
        >       Jenkinsfile
        > module1
        > Module2
        > Module3
        > Module4
        > terraform_level3.pdf
        > tf-adv-lab1-feb26
        > tf-adv-lab1-feb26-b2
        > tf-adv-sca-feb26
        > tf-adv-secrets
        >   > infra
        >     Jenkinsfile
        > .bash_history
    
```

```

64 // }
65 // stage("che
66 // steps {
67 // sh """
68 // dock

PROBLEMS OUTPUT DEBUG

● [ec2-user@ip-172-31-33-63
Cloning into 'tf-adv-secr
warning: You appear to ha
● [ec2-user@ip-172-31-33-63
bash: cd: tf-adv-: No suc
● [ec2-user@ip-172-31-33-63
● [ec2-user@ip-172-31-33-63
● [ec2-user@ip-172-31-33-63
. . . .git
○ [ec2-user@ip-172-31-33-63

```

#### 4. Configure the secret mgr name in the tf main file

```

EXPLORER ... Jenkinsfile backend.tf ~.../Lab02-Solution/...
EC2-USER
  > tf-adv-im-feb26
    > Module3
    > Module4
    > terraform_level3.pdf
    > tf-adv-lab1-feb26
    > tf-adv-lab1-feb26-b2
    > tf-adv-sca-feb26
    > tf-adv-secrets
      > infra
        > backend.tf
        > main.tf
        > userdata.sh
        Jenkinsfile

```

```

tf-adv-secrets > infra > main.tf > resource "aws_instance" {
  provider "aws" {
    region = "us-west-2"
  }
  data "aws_secretsmanager_secret_version" "a
  secret_id = "dev/app/api_key"
}
resource "aws_instance" "secure_app" {
  ami           = "ami-0c02fb55956c7d316"
  instance_type = "t3.micro"
  user_data = templatefile("userdata.sh",
  api_key = data.aws_secretsmanager_secre
}

```

#### 5. Git commands to push it

```

. . . .git
● [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ git add .
● [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ git commit -m "secerts"
[main (root-commit) 436a9a9] secerts
 4 files changed, 103 insertions(+)
  create mode 100644 Jenkinsfile
  create mode 100644 infra/backend.tf
  create mode 100644 infra/main.tf
  create mode 100644 infra/userdata.sh
● [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.28 KiB | 1.28 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:Vishwanathms/tf-adv-secrets.git
 * [new branch]      main -> main
○ [ec2-user@ip-172-31-33-63 tf-adv-secrets]$ 

```

Validated on the github portal

Vishwanathms / **tf-adv-secrets**

Type  to search

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

**tf-adv-secrets** Public

[Pin](#) [Watch 0](#)

[main](#) [1 Branch](#) [0 Tags](#)

Go to file [t](#) [+](#) [Code](#)

vishwa	secerts	436a9a9 · 3 minutes ago	1 Commit
infra	secerts	3 minutes ago	
Jenkinsfile	secerts	3 minutes ago	

[README](#)

Add a README

Help people interested in this repository understand your project.

[Add a README](#)

## 6. Create Jenkins job for secrets

## New Item

Enter an item name

vish-job3-secrets

Select an item type



Pipeline

Build, test, and deploy using pipelines. !



Freestyle project

Classic, general-purpose job type that c  
steps like archiving artifacts and sendin



Multi-configuration project

Suitable for projects that need a large r  
platform-specific builds, etc.



Folder

Creates a container that stores nested i  
folder creates a separate namespace, sc  
folders.



Multibranch Pipeline

Creates a set of Pipeline projects accor



Organization Folder

Creates a set of multibranch project sut

If you want to create a new item from other existing

Copy from

OK



## Configure

## General

 General Triggers Pipeline Advanced

## Description

---

---

Plain text [Preview](#)

- Discard old builds ?
- Do not allow concurrent builds
- Do not allow the pipeline to resume if the controller restarts
- GitHub project
- Pipeline speed/durability override ?
- Preserve stashes from completed builds ?
- This project is parameterized ?
- Throttle builds ?

 Pipeline speed/durability override

- Preserve stashes from completed builds ?

 This project is parameterized ?

## String Parameter ?

## Name ?

branchname

## Default Value ?

main

## Description ?

select an branch

Plain text [Preview](#)

- Trim the string ?

S / Configure

Pipeline script from SCM

Definition

SCM ?

Git

Repositories ?

Repository URL ?

`https://github.com/Vishwanathms/tf-adv-secrets.git`

Credentials ?

- none -

Advanced ▾

General

Triggers

**Pipeline**

Advanced

Branch Specifier (blank for 'any') ?

`*/main`

+ Add Branch

Repository browser ?

(Auto)

Additional Behaviours

+ Add

Script Path ?

`Jenkinsfile`

Lightweight checkout ?

Pipeline Syntax

Advanced

Save Apply

The screenshot shows the Jenkins Pipeline configuration screen. The 'Pipeline' tab is selected. Under the 'SCM' section, 'Git' is chosen. The 'Repository URL' is set to 'https://github.com/Vishwanathms/tf-adv-secrets.git'. The 'Branch Specifier' is set to '\*/main'. The 'Script Path' is set to 'Jenkinsfile'. The 'Lightweight checkout' checkbox is checked. At the bottom, there are 'Save' and 'Apply' buttons.

7. Run the job.

```
time          w1.com
  or timezone

[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Terraform plan)
[Pipeline] sh
02:27:07 + cd infra
02:27:07 + terraform plan -out=tfplan
02:27:13 0m [1mdata.aws_secretsmanager_secret_version.api: Reading...0m0m
02:27:13 0m [1mdata.aws_secretsmanager_secret_version.api: Read complete after 0s [id=dev/app/api_key|AWSCURRENT]0m
02:27:13
02:27:13 Terraform used the selected providers to generate the following execution
02:27:13 plan. Resource actions are indicated with the following symbols:
02:27:13 0m [32m+0m create0m
02:27:13
02:27:13 Terraform will perform the following actions:
02:27:13
02:27:13 0m # aws_instance.secure_app0m will be created
02:27:13 0m 0m [32m+0m0m resource "aws_instance" "secure_app" {
02:27:13 0m 0m [32m+0m0m ami           = "ami-0c02fb55956c7d316"
02:27:13 0m 0m [32m+0m0m arn          = "(known after apply)"
02:27:13 0m 0m [32m+0m0m associate_public_ip_address = "(known after apply)"
02:27:13 0m 0m [32m+0m0m availability_zone   = "(known after apply)"
02:27:13 0m 0m [32m+0m0m disable_api_stop    = "(known after apply)"
02:27:13 0m 0m [32m+0m0m disable_api_termination = "(known after apply)"
02:27:13 0m 0m [32m+0m0m ebs_optimized      = "(known after apply)"
02:27:13 0m 0m [32m+0m0m enable_primary_ipv6     = "(known after apply)"
```

This confirm the secrets mgr value was able to be read by the terraform