

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

AWS Secrets Manager > Secrets > Store a new secret

Step 1: Choose secret type (selected)
Step 2: Configure secret
Step 3: optional: Configure rotation
Step 4: Review

Choose secret type

Secret type Info

☐ Credentials for Amazon RDS database

☐ Credentials for Amazon DocumentDB database

☐ Credentials for Amazon Redshift data warehouse

☐ Credentials for other database

☐ Managed external secret
Secrets vendored by your third-party software vendor.

☒ Other type of secret
API key, OAuth token, other.

Key/value pairs

 Info

Key/value Plaintext

| | |
|---------|--------------|
| api_key | Password123! |
|---------|--------------|

+ Add row

Encryption key

 Info

You can encrypt using the KMS key that Secrets Manager creates or a customer-managed KMS key that you create.

aws/secretsmanager

Add new key

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](#) [Refresh](#)

[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.

Disable secret and schedule deletion

You are attempting to delete the primary secret **prod/app/api_key** in **us-west-2**. The last time this secret was accessed was **0 days** ago. Secrets Manager requires a minimum waiting period of 7 days before deleting a secret. You will not be able to retrieve the secret once it is scheduled for deletion.

Waiting period
 days
 Must be between 7 and 30 days.

Cancel **Schedule deletion**

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

tf-adv-secrets is available.

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

Description

tf-adv-secrets

14 / 350 characters

2

Configuration

Choose visibility *

Choose who can see and commit to this repository

Public

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

Off

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore

Add license

Licenses explain how others can use your code. [About licenses](#)

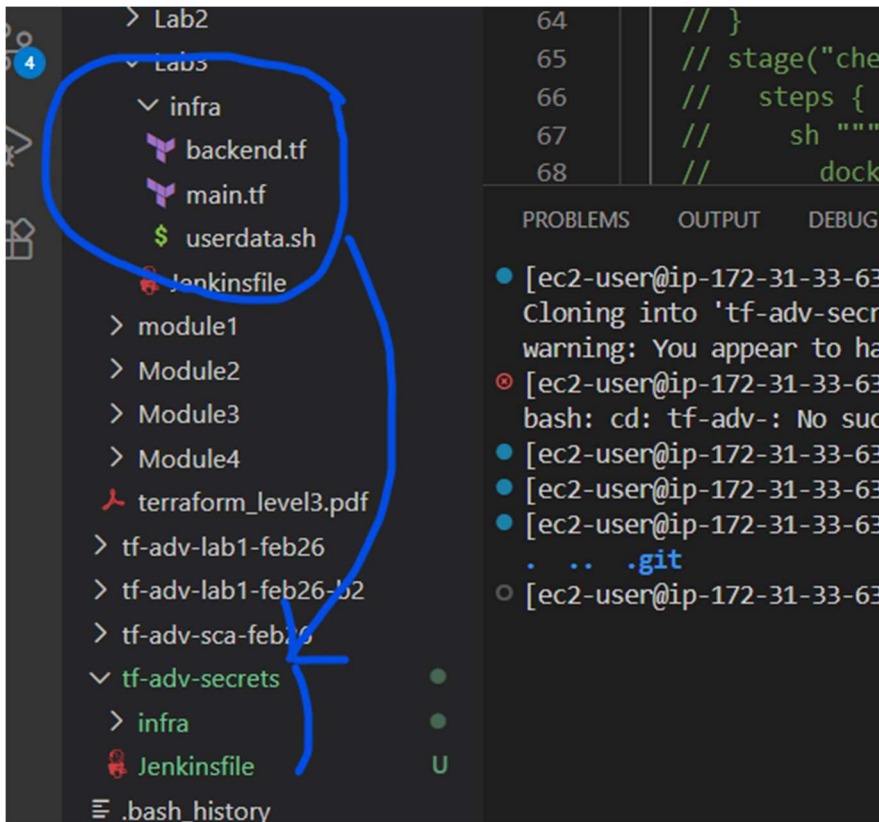
No license

Create repository

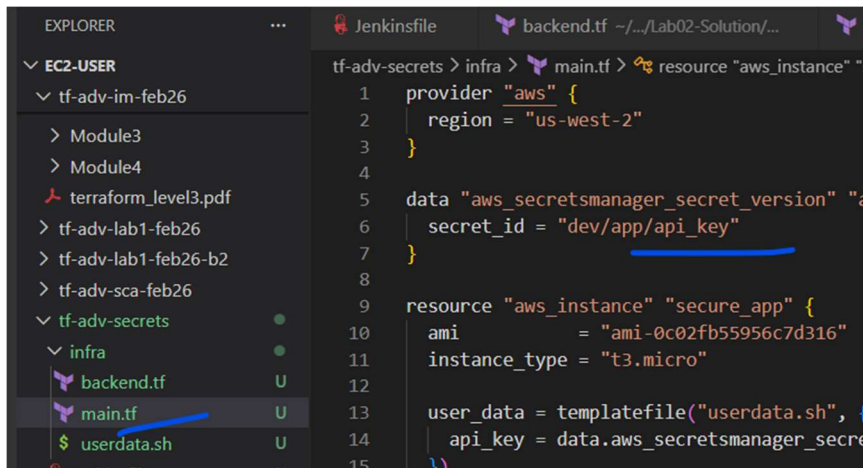
```
[ec2-user@ip-172-31-33-63 ~]$ cd tf-adv-secrets/
[ec2-user@ip-172-31-33-63 tf-adv-secrets]$ ls
[ec2-user@ip-172-31-33-63 tf-adv-secrets]$ ls -la
.  ..  .git
[ec2-user@ip-172-31-33-63 tf-adv-secrets]$
```

Cloned on the vs code.

3. Copy the lab 3 code to this repo folder



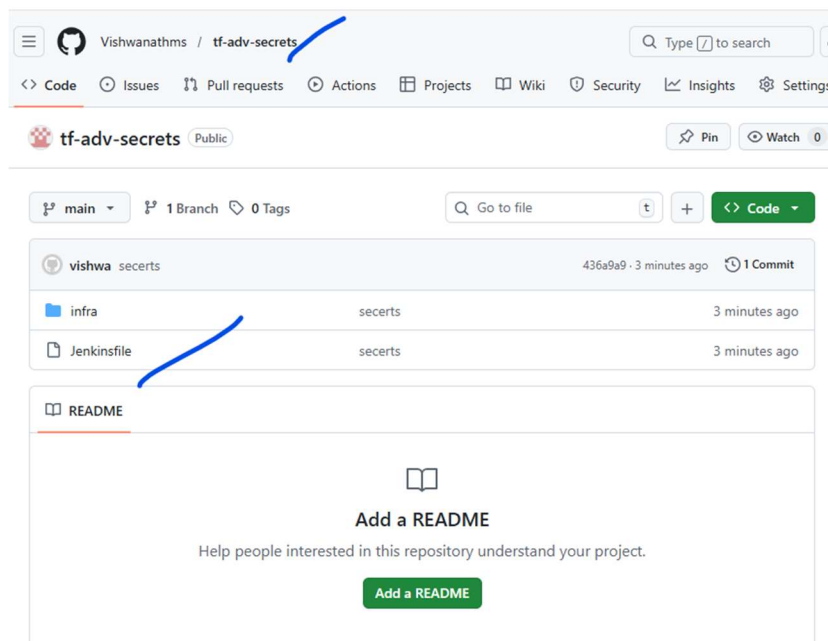
4. Configure the secret mgr name in the tf main file



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




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. [Learn more](#)
- 

Freestyle project
Classic, general-purpose job type that can perform a wide range of steps like archiving artifacts and sending notifications.
- 

Multi-configuration project
Suitable for projects that need a large number of builds, such as platform-specific builds, etc.
- 

Folder
Creates a container that stores nested items. A folder creates a separate namespace, so items within the folder are not visible to other users.
- 

Multibranch Pipeline
Creates a set of Pipeline projects according to a given pattern.
- 

Organization Folder
Creates a set of multibranch project sub-items.

If you want to create a new item from other existing item

Copy from

OK



Configure

General

Triggers

Pipeline

Advanced

General

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

Permission

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

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

Credentials ?

- none -

Advanced ▾

Branch Specifier (blank for 'any') ?

`*/main`

+ Add Branch

Repository browser ?

(Auto)

Additional Behaviours

+ Add

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Advanced

Advanced ▾

Save Apply

7. Run the job.

time
or timezone

```
02:27:07 [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...[0m[0m
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   [32m+[0m create[0m
02:27:13
02:27:13 Terraform will perform the following actions:
02:27:13
02:27:13 [1m # aws_instance.secure_app[0m will be created
02:27:13 [0m [32m+[0m resource "aws_instance" "secure_app" {
02:27:13   [32m+[0m ami                               = "ami-0c02fb55956c7d316"
02:27:13   [32m+[0m arn                               = (known after apply)
02:27:13   [32m+[0m associate_public_ip_address       = (known after apply)
02:27:13   [32m+[0m availability_zone                   = (known after apply)
02:27:13   [32m+[0m disable_api_stop                   = (known after apply)
02:27:13   [32m+[0m disable_api_termination             = (known after apply)
02:27:13   [32m+[0m ebs_optimized                       = (known after apply)
02:27:13   [32m+[0m enable_primary_ipv6                 = (known after apply)
```

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