

## Terraform State file Practice

The screenshot shows a VS Code editor with a Terraform configuration file named `backend.tf`. The configuration is as follows:

```
1 terraform {
2   backend "s3" {
3     bucket = "bucket26032025"
4     key    = "state_file/terraform.state"
5     region = "ap-south-1"
6     dynamodb_table = "terraform-state-locks-3jul"
7   }
8 }
```

The Explorer sidebar on the left shows the project structure with files like `provider.tf`, `resource.tf`, `s3.tf`, `terraform.tfstate`, and `backend.tf`. The terminal at the bottom shows the output of the `Terraform init` command:

```
Apply complete! Resources: 1 added, 0 changed, 1 destroyed.
PS C:\Users\DELL\OneDrive\Desktop\git\Terraform> Terraform init
Initializing the backend...
Do you want to copy existing state to the new backend?
Pre-existing state was found while migrating the previous "local" backend to the
newly configured "s3" backend. No existing state was found in the newly
configured "s3" backend. Do you want to copy this state to the new "s3"
backend? Enter "yes" to copy and "no" to start with an empty state.

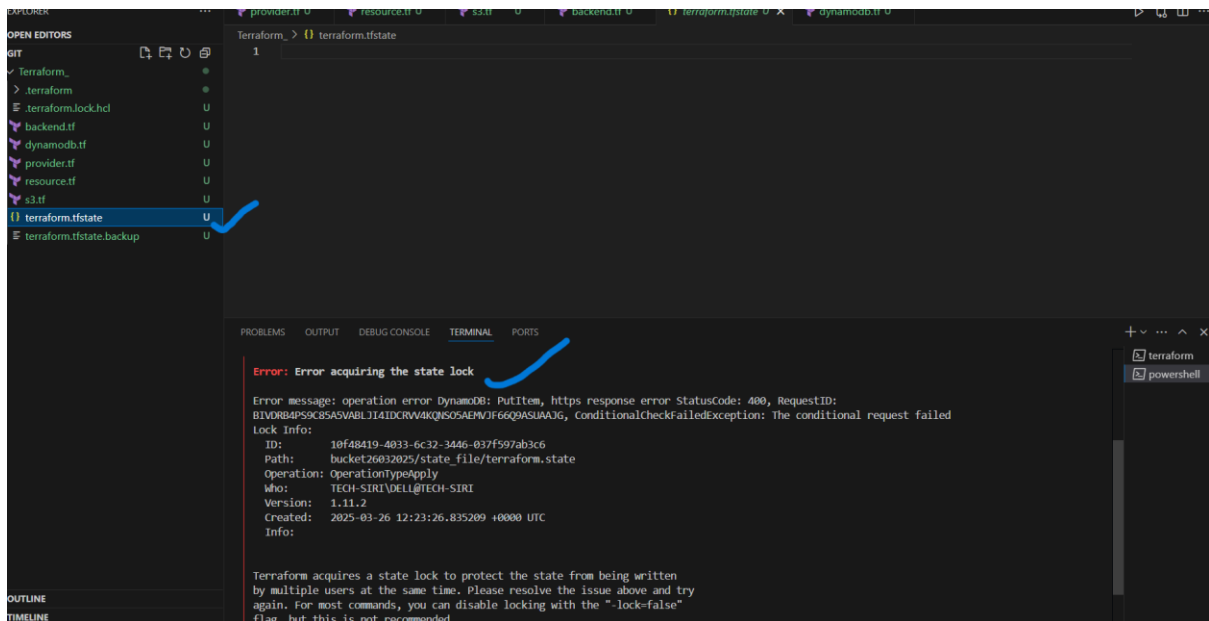
Enter a value: yes

Releasing state lock. This may take a few moments...

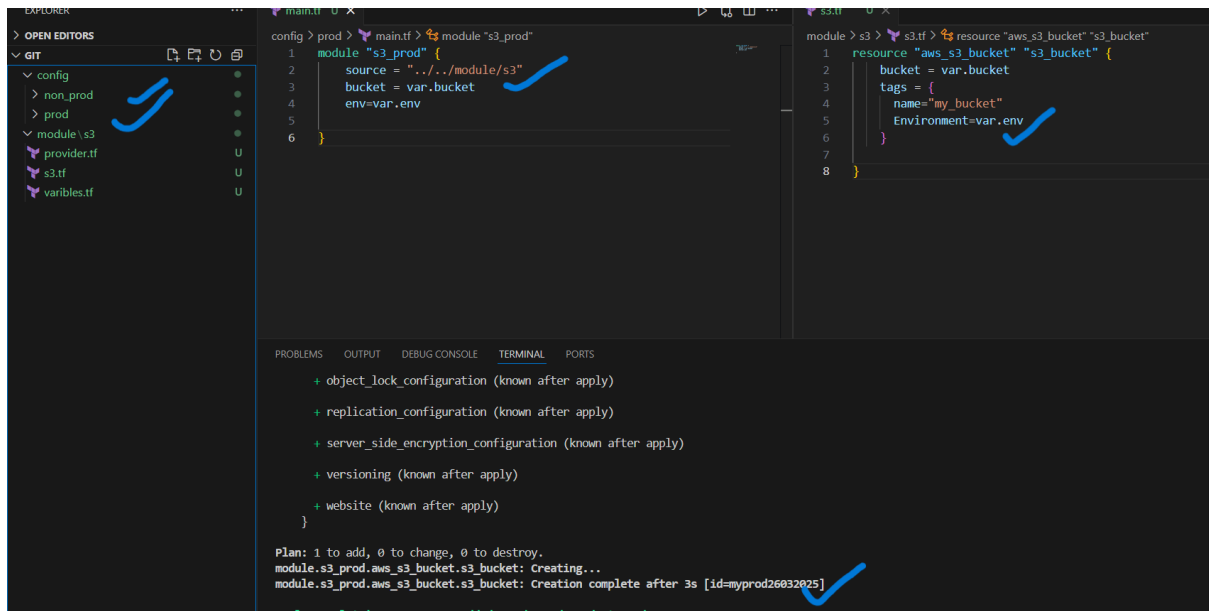
Successfully configured the backend "s3"! Terraform will automatically
use this backend unless the backend configuration changes.
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
```

The screenshot shows the AWS S3 console interface. The breadcrumb navigation indicates the path: `Buckets > bucket26032025 > state_file/`. The `state_file/` folder is selected, showing a single object named `terraform.state` with a size of 10.2 KB and a storage class of Standard. The object was last modified on March 26, 2025, at 17:53:04 (UTC+05:30).

Below the S3 console, the AWS DynamoDB console is shown, displaying a table named `terraform-state-locks-3jul`. The table is in an `Active` state and uses the `LockID (S)` as the partition key. The table has 0 indexes and 0 items.



## Module Practice



### Account snapshot - updated every 24 hours All AWS Regions

[View Storage Lens dashboard](#)

Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

### General purpose buckets

### Directory buckets

### General purpose buckets (2) Info All AWS Regions

Buckets are containers for data stored in S3.



[Copy ARN](#)

[Empty](#)

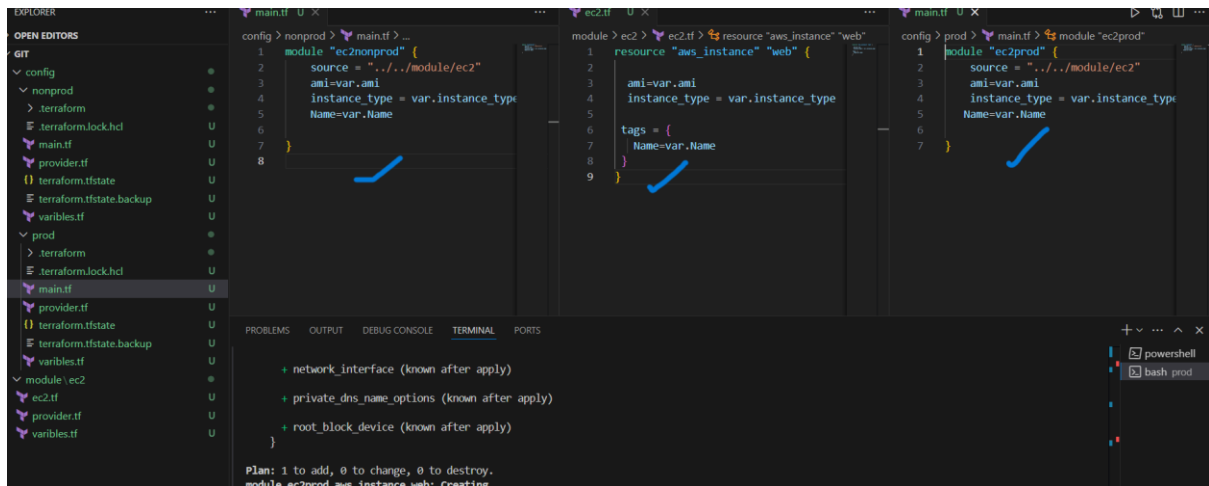
[Delete](#)

[Create bucket](#)

Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/> <a href="#">mynonprod26032025</a>	Asia Pacific (Mumbai) ap-south-1	<a href="#">View analyzer for ap-south-1</a>	March 26, 2025, 20:06:32 (UTC+05:30)
<input type="radio"/> <a href="#">myprod26032025</a>	Asia Pacific (Mumbai) ap-south-1	<a href="#">View analyzer for ap-south-1</a>	March 26, 2025, 20:11:52 (UTC+05:30)

## LAB ACTIVITY

1. provision EC2 instances using modules, if it is prod environment, provision t2.medium instance, if it is non prod environment provision t2.micro instance here reference source should be pointing to local.



Instances (2) Info

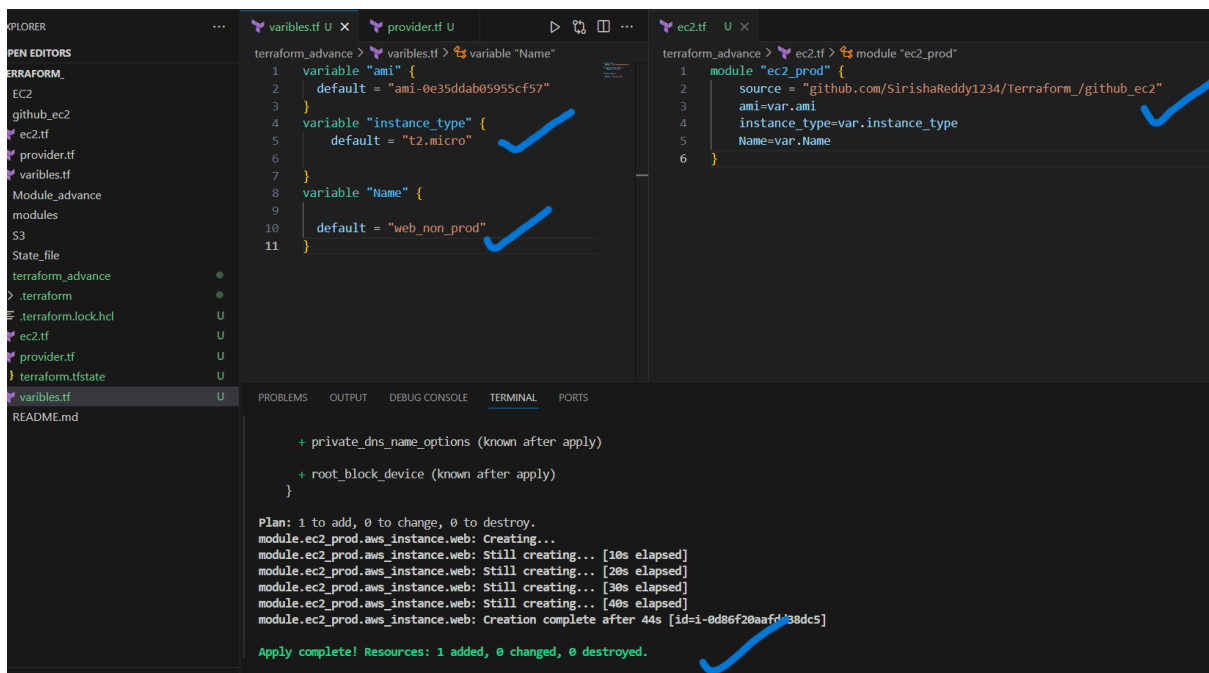
Last updated 2 minutes ago

Find Instance by attribute or tag (case-sensitive)

Instance state = running

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
web_prod	i-0367f60eb5d27306a	Running	t2.medium	Initializing	View alarms +	ap-south-1a	ec2-13-233
web_non_prod	i-042cbf2aea0472b72	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1b	ec2-3-110-2

Create an EC2 instance through modules code and push them GitHub repo, In root module while calling child modules, source you should select from github and deploy an EC2 instance on non prod.



All states

< 1 >

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public
<input type="checkbox"/>	web_non_prod	i-0c627fd166ba99a0b	Running	t2.micro	...	View alarms +	ap-south-1b	ec2-13

Create IAM User while source argument point to terraform registry.

```

main.tf U X
iam_terraform_regi > main.tf > module "iam_user"
1  module "iam_user" {
2      source = "terraform-aws-modules/iam/aws//modules/iam-user"
3
4      name = "vasya.pupkin"
5      force_destroy = true
6
7      pgp_key = "keybase:test"
8
9      password_reset_required = false
10 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

+ password_length      = 20
+ password_reset_required = false
+ pgp_key              = "keybase:test"
+ user                 = "vasya.pupkin"
}

Plan: 3 to add, 0 to change, 0 to destroy.
module.iam_user.aws_iam_user.this[0]: Creating...
module.iam_user.aws_iam_user.this[0]: Creation complete after 2s [id=vasya.pupkin]
module.iam_user.aws_iam_user_login_profile.this[0]: Creating...
module.iam_user.aws_iam_access_key.this[0]: Creating...
module.iam_user.aws_iam_user_login_profile.this[0]: Creation complete after 2s [id=vasya.pupkin]
module.iam_user.aws_iam_access_key.this[0]: Creation complete after 2s [id=AKIAWQUOZX2BBQ0HITJE]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
DELL@TECH-SIRI MINGW64 ~/OneDrive/Desktop/github/Terraform/Iam_terraform_regi (main)

```

GG

## Users (2) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Search							
<input type="checkbox"/>	User name	Path	Group	Last activity	MFA	Password age	Cons
<input type="checkbox"/>	siri-terra	/	0	16 minutes ago	-	6 days	-
<input type="checkbox"/>	vasya.pupkin	/	0	-	-	-	-

GitHub repository : [https://github.com/SirishaReddy1234/Terraform\\_.git](https://github.com/SirishaReddy1234/Terraform_.git)