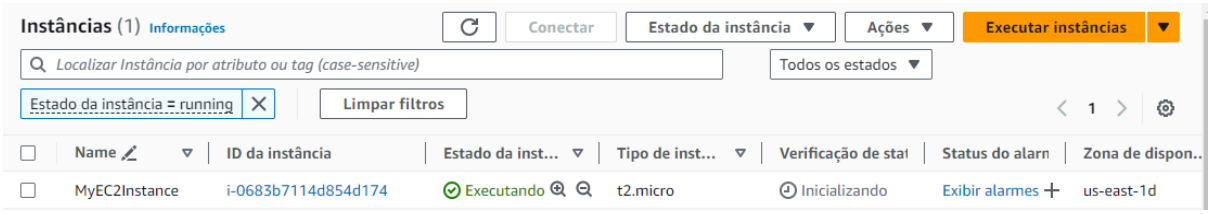
```
felipe@Felipe:~$ sudo apt-get install terraform
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 19 not upgraded.
Need to get 28.0 MB of archives.
After this operation, 89.0 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com jammy/main amd64 terraform amd64 1.9.1-1 [28.0 MB]
Fetched 28.0 MB in 6s (4662 kB/s)
Selecting previously unselected package terraform.
(Reading database ... 44555 files and directories currently installed.)
Preparing to unpack .../terraform_1.9.1-1_amd64.deb ...
Unpacking terraform (1.9.1-1) ...
Setting up terraform (1.9.1-1) ...
felipe@Felipe:~$ terraform -v
Terraform v1.9.1
on linux_amd64
felipe@Felipe:~$
```

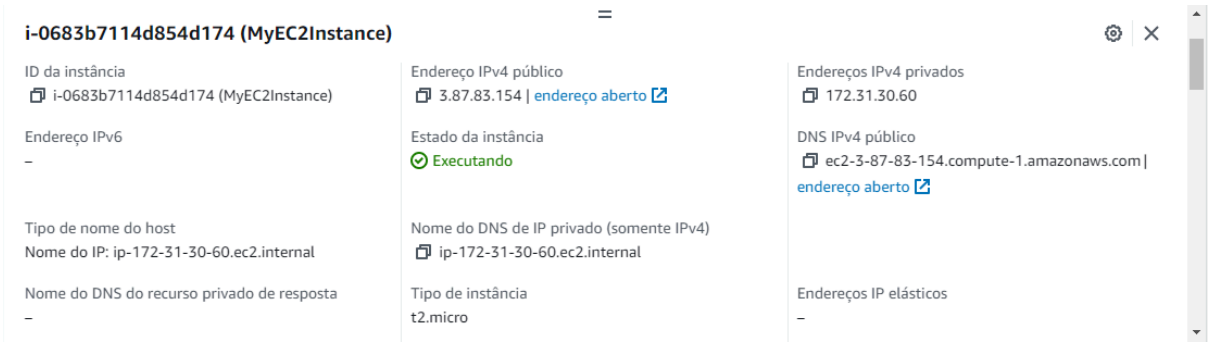
```
felipe@Felipe:~/terraform-aws-setup$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v5.57.0...
- Installed hashicorp/aws v5.57.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
felipe@Felipe:~/terraform-aws-setup$
```

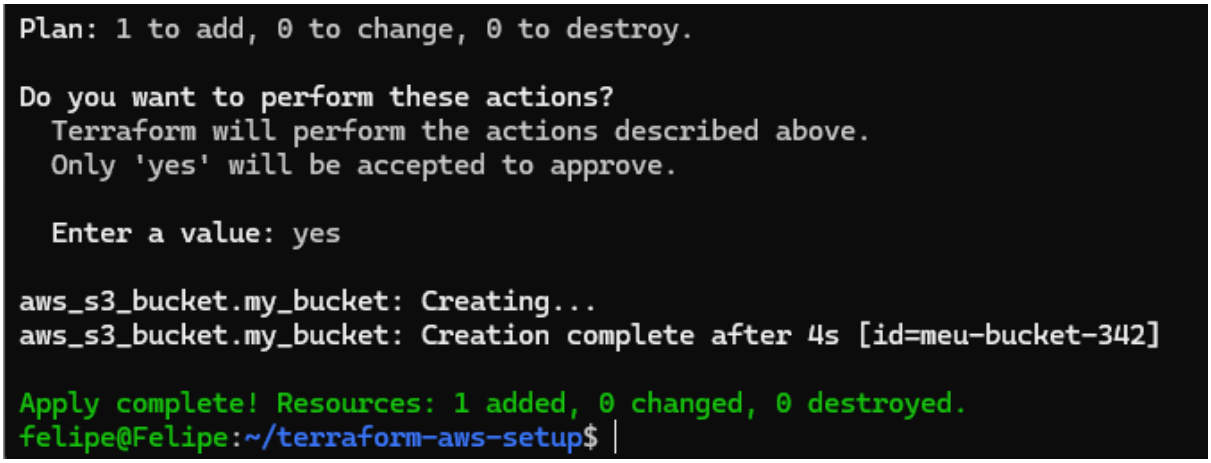
```
      }
    + website_domain              = (known after apply)
    + website_endpoint            = (known after apply)

    + cors_rule (known after apply)

    + grant (known after apply)

    + lifecycle_rule (known after apply)

    + logging (known after apply)

    + object_lock_configuration (known after apply)

    + replication_configuration (known after apply)

    + server_side_encryption_configuration (known after apply)

    + versioning (known after apply)

    + website (known after apply)
  }

Plan: 2 to add, 0 to change, 0 to destroy.

─────────────────────────────────────────────────────────────────────────────

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if
you run "terraform apply" now.
felipe@Felipe:~/terraform-aws-setup$
```

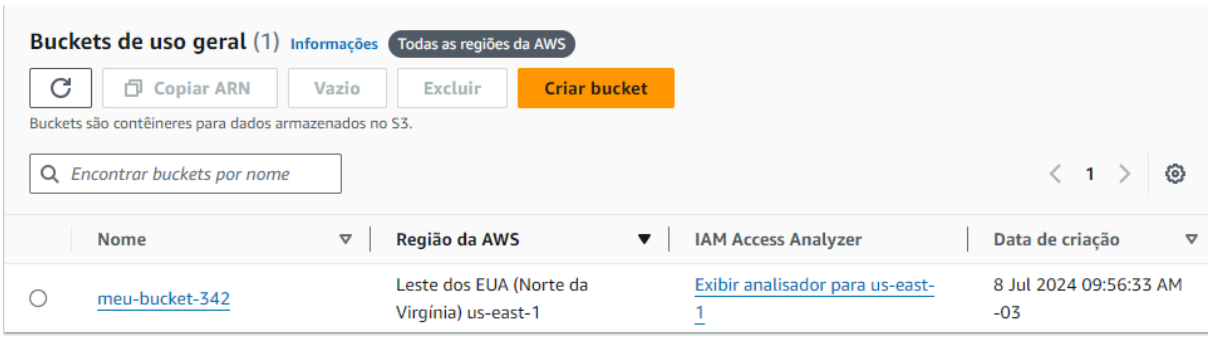
```
felipe@Felipe: ~/terraform-aw
    + cors_rule (known after apply)

    + grant (known after apply)

    + lifecycle_rule (known after apply)

    + logging (known after apply)

    + object_lock_configuration (known after apply)

    + replication_configuration (known after apply)

    + server_side_encryption_configuration (known after apply)

    + versioning (known after apply)

    + website (known after apply)
  }

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.my_ec2: Creating...
aws_s3_bucket.my_bucket: Creating...
```

```
Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.my_ec2: Creating...
aws_s3_bucket.my_bucket: Creating...
aws_instance.my_ec2: Still creating... [10s elapsed]
aws_instance.my_ec2: Still creating... [20s elapsed]
aws_instance.my_ec2: Still creating... [30s elapsed]
aws_instance.my_ec2: Creation complete after 35s [id=i-0683b7114d854d174]
```


Instâncias (1) Informações
Conectar
Estado da instância
Ações
Executar instâncias
Localizar Instância por atributo ou tag (case-sensitive)
Todos os estados
Estado da instância = running
Limpar filtros
1
Name
ID da instância
Estado da inst...
Tipo de inst...
Verificação de stat
Status do alarn
Zona de dispon..
MyEC2Instance
i-0683b7114d854d174
Executando
t2.micro
Inicializando
Exibir alarmes
us-east-1d


i-0683b7114d854d174 (MyEC2Instance)
ID da instância
i-0683b7114d854d174 (MyEC2Instance)
Endereço IPv4 público
3.87.83.154 | endereço aberto
Endereços IPv4 privados
172.31.30.60
Endereço IPv6
–
Estado da instância
Executando
DNS IPv4 público
ec2-3-87-83-154.compute-1.amazonaws.com | endereço aberto
Tipo de nome do host
Nome do IP: ip-172-31-30-60.ec2.internal
Nome do DNS de IP privado (somente IPv4)
ip-172-31-30-60.ec2.internal
Nome do DNS do recurso privado de resposta
–
Tipo de instância
t2.micro
Endereços IP elásticos
–


Plan: 1 to add, 0 to change, 0 to destroy.
Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.
Enter a value: yes
aws_s3_bucket.my_bucket: Creating...
aws_s3_bucket.my_bucket: Creation complete after 4s [id=meu-bucket-342]
Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
felipe@Felipe:~/terraform-aws-setup$


Buckets de uso geral (1) Informações
Todas as regiões da AWS
Copiar ARN
Vazio
Excluir
Criar bucket
Buckets são contêineres para dados armazenados no S3.
Encontrar buckets por nome
1
Nome
Região da AWS
IAM Access Analyzer
Data de criação
meu-bucket-342
Leste dos EUA (Norte da Virgínia) us-east-1
Exibir analisador para us-east-1
8 Jul 2024 09:56:33 AM -03