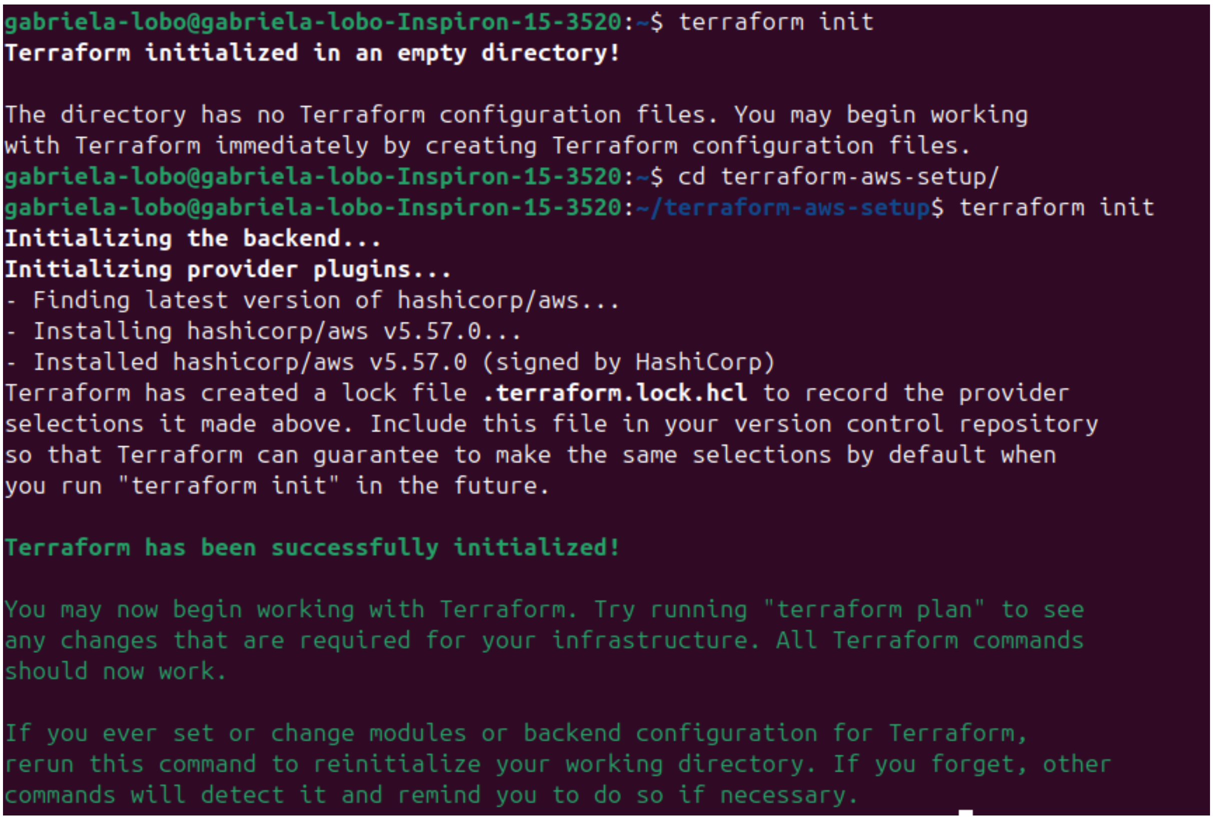
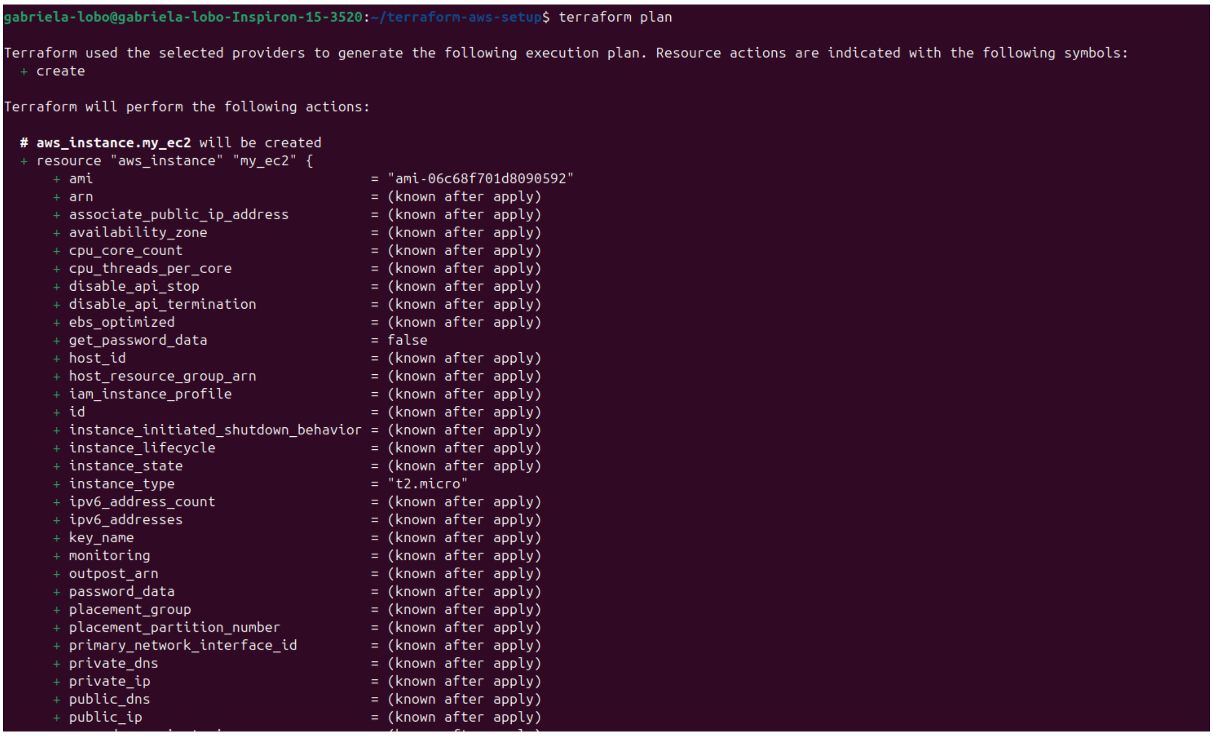
Terraform init

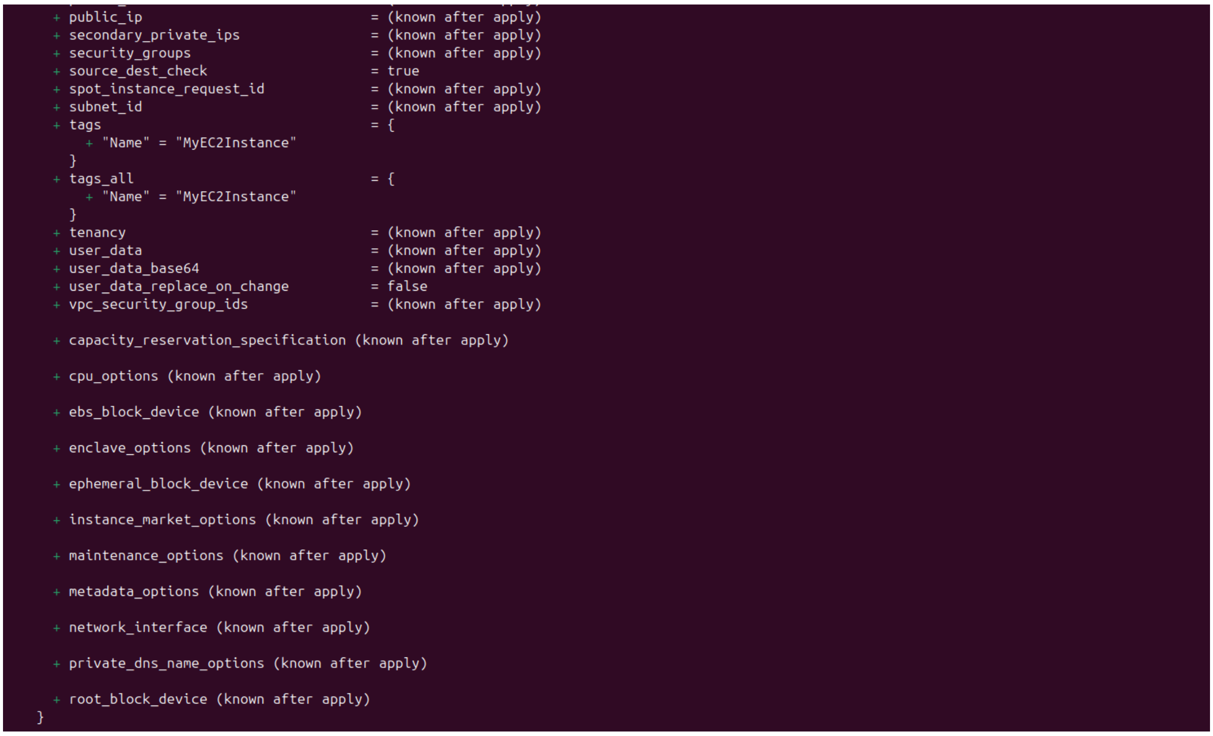


```
gabriela-lobo@gabriela-lobo-Inspiron-15-3520:~$ terraform init
Terraform initialized in an empty directory!

The directory has no Terraform configuration files. You may begin working
with Terraform immediately by creating Terraform configuration files.
gabriela-lobo@gabriela-lobo-Inspiron-15-3520:~$ cd terraform-aws-setup/
gabriela-lobo@gabriela-lobo-Inspiron-15-3520:~/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.
```

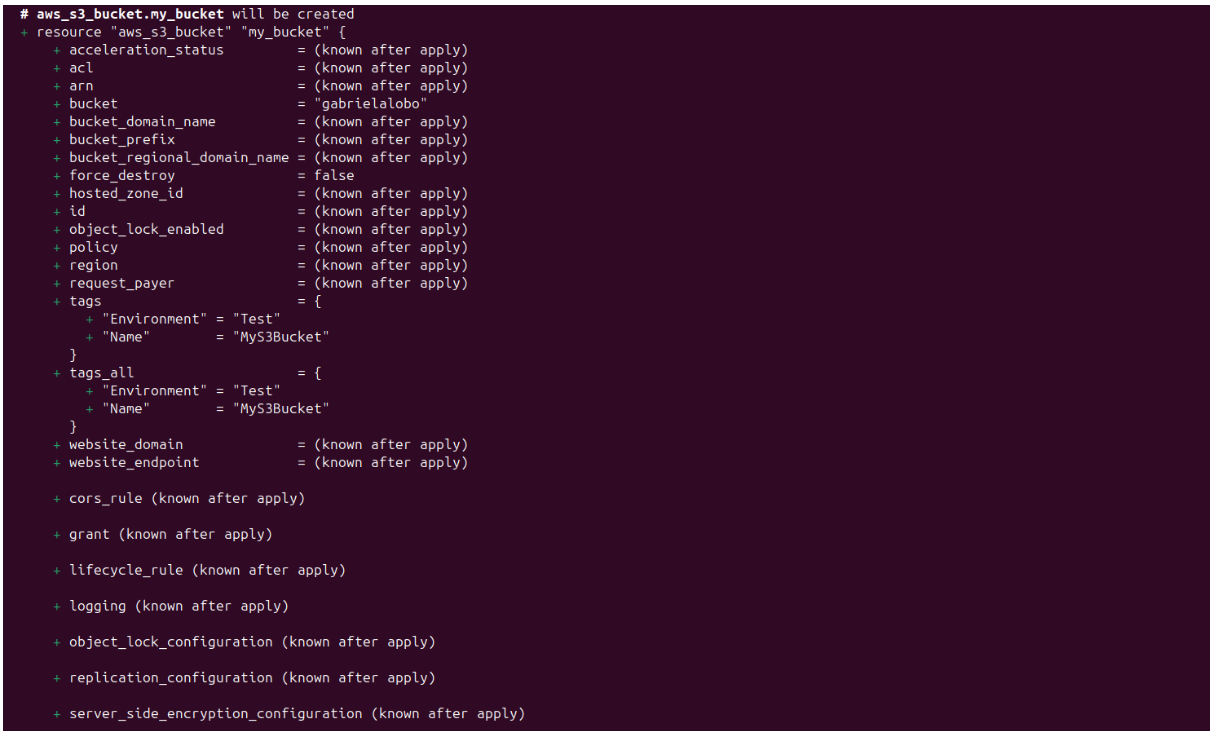
Terraform plan



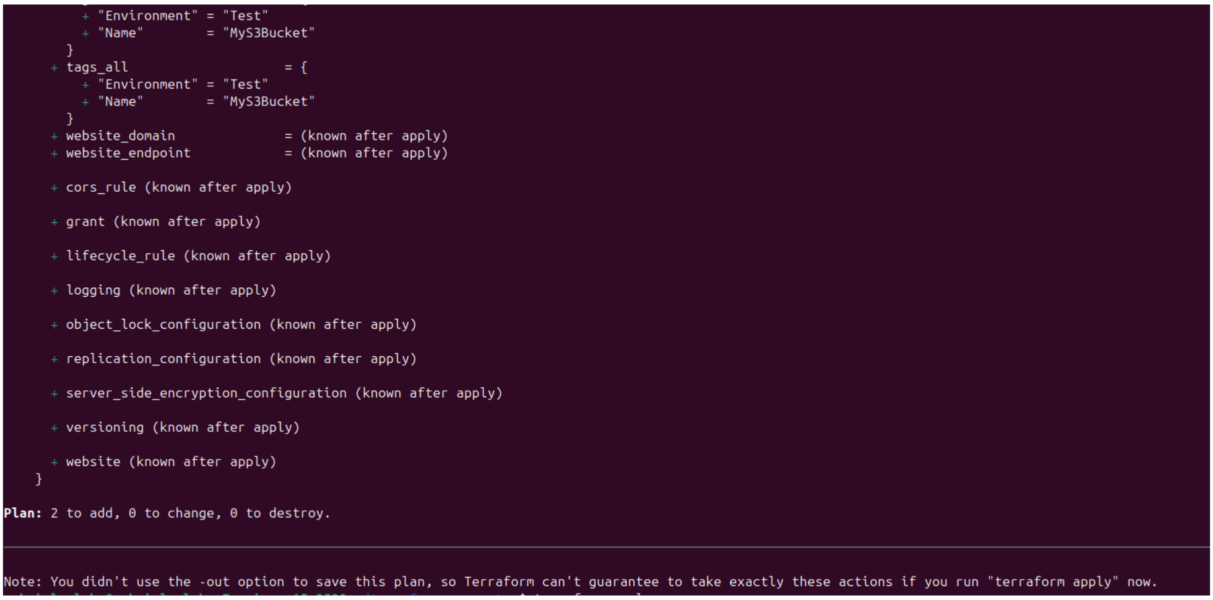
```
gabriela-lobo@gabriela-lobo-Inspiron-15-3520:~/terraform-aws-setup$ terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.my_ec2 will be created
  + resource "aws_instance" "my_ec2" {
      + ami                                  = "ami-06c68f701d8090592"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
```



```
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "MyEC2Instance"
        }
      + tags_all                             = {
          + "Name" = "MyEC2Instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }
```

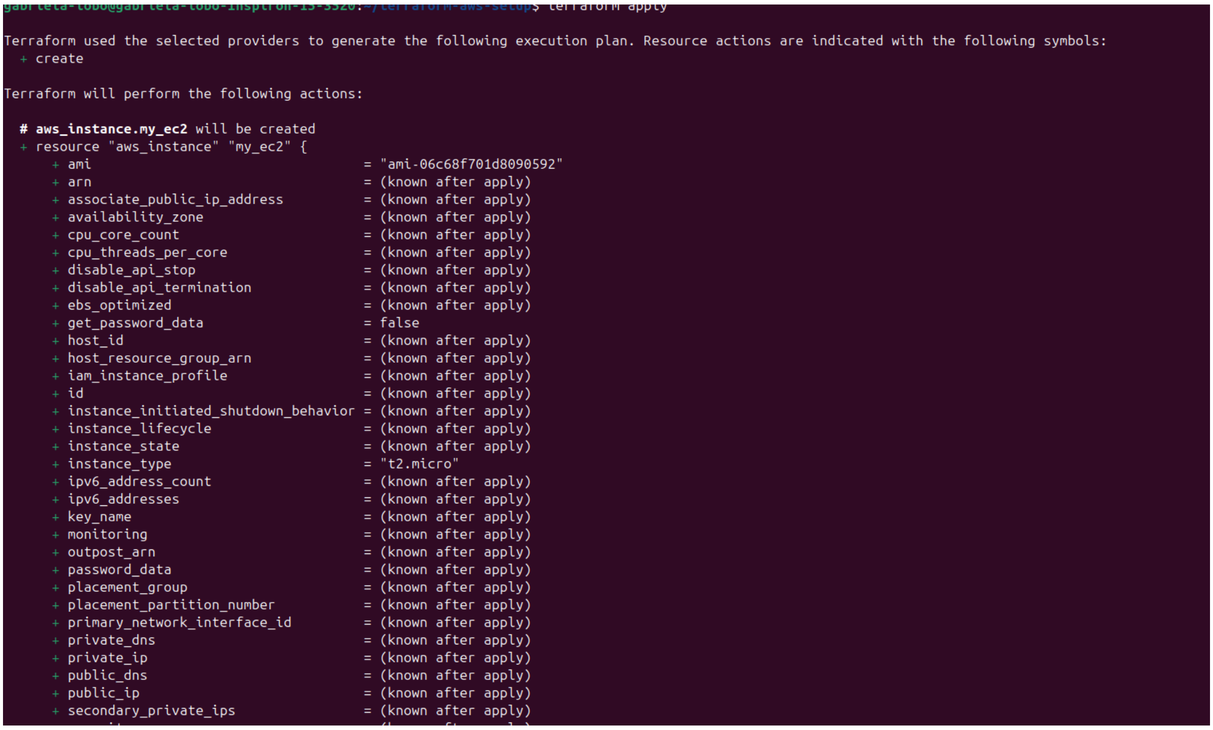


```
  # aws_s3_bucket.my_bucket will be created
  + resource "aws_s3_bucket" "my_bucket" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "gabrielalobo"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags                        = {
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + tags_all                    = {
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + 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)
```

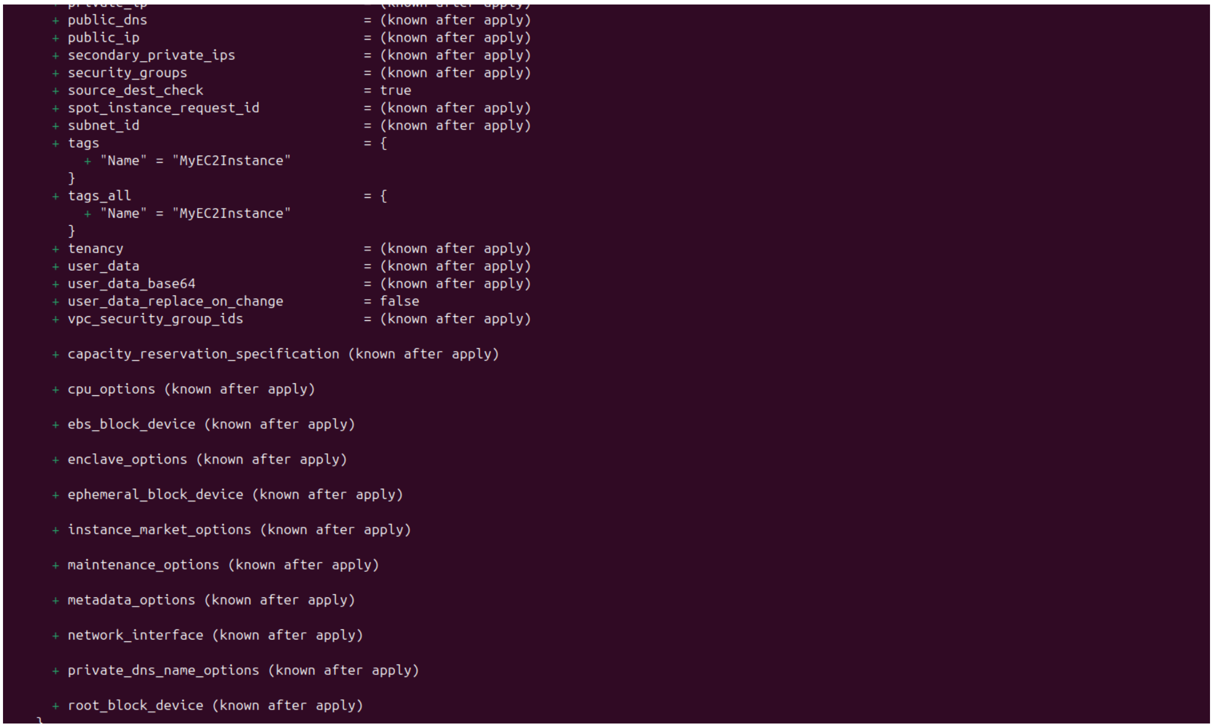


```
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + tags_all                    = {
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + website_domain              = (known after apply)
      + website_endpoint            = (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.
```

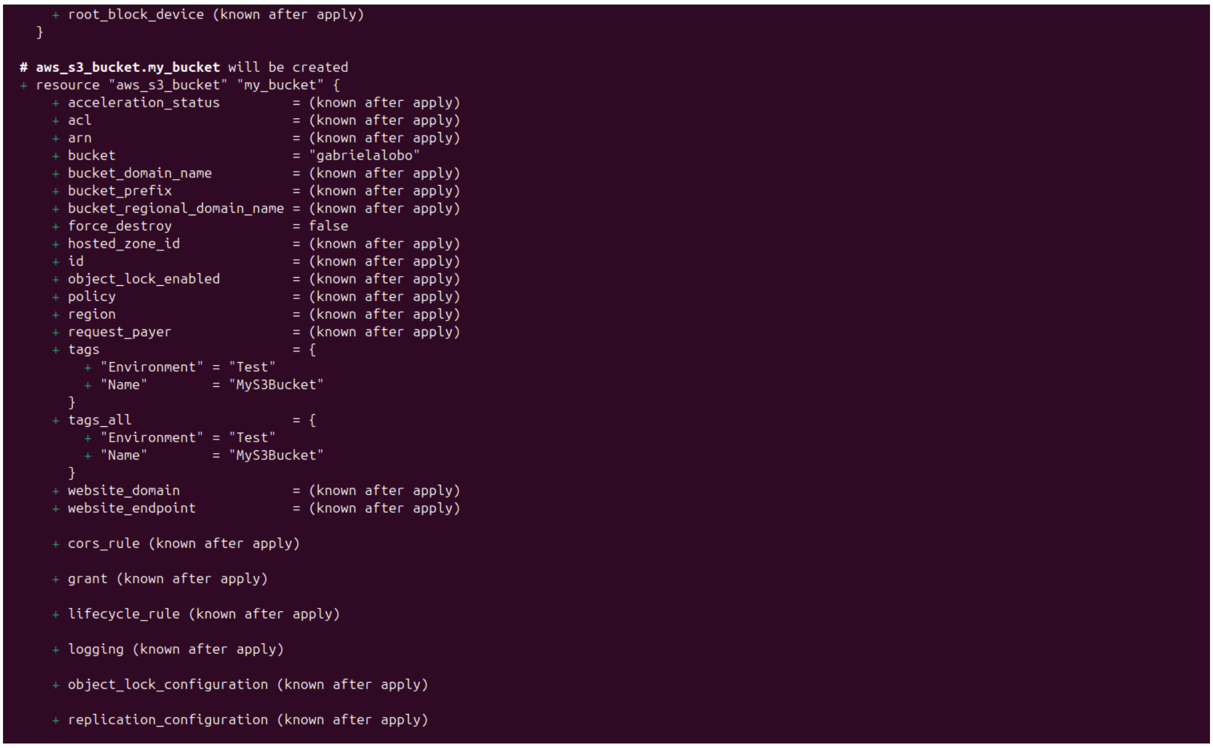
Terraform Apply



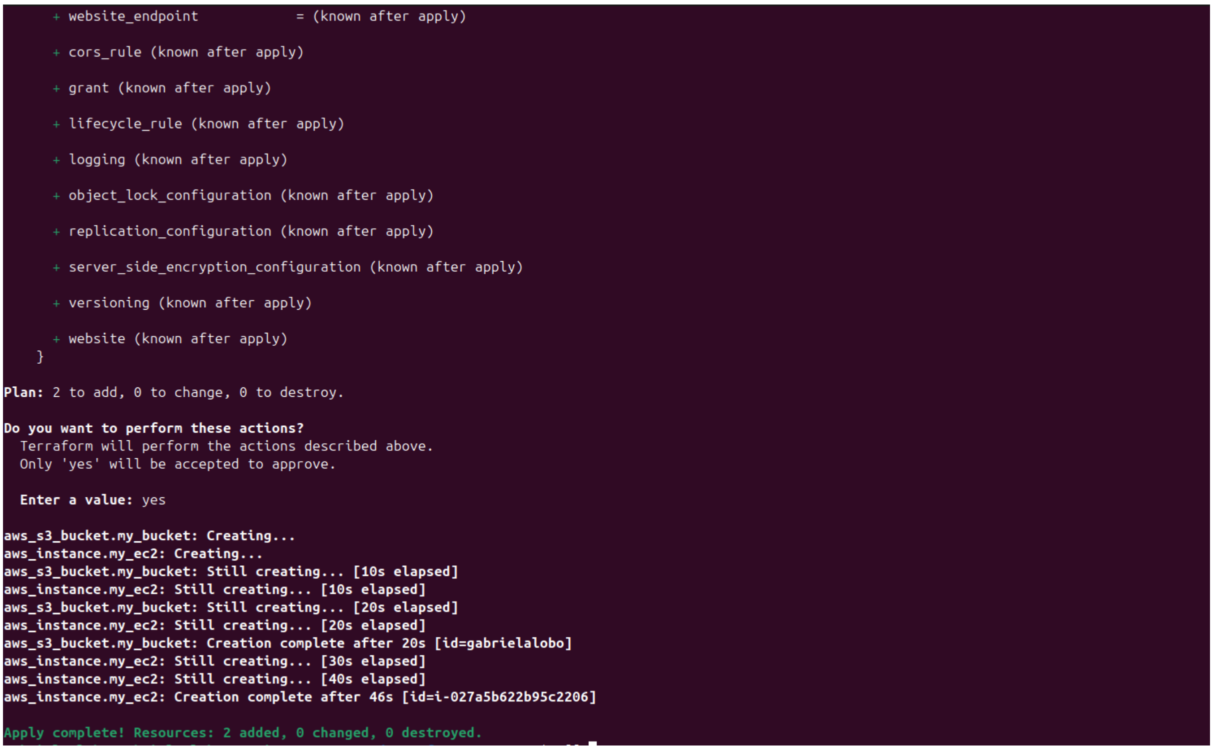
```
gabriela-lobo@gabriela-lobo-Inspiron-15-3520:~/terraform-aws-setup$ terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.my_ec2 will be created
  + resource "aws_instance" "my_ec2" {
      + ami                                  = "ami-06c68f701d8090592"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
```



```
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "MyEC2Instance"
        }
      + tags_all                             = {
          + "Name" = "MyEC2Instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }
```

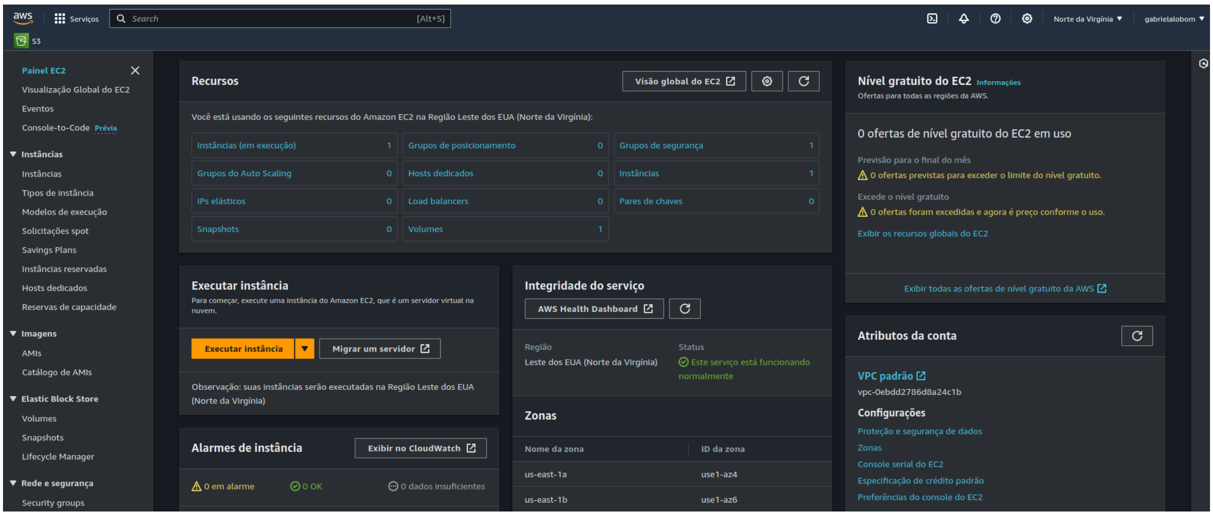


```
      + root_block_device (known after apply)
    }

  # aws_s3_bucket.my_bucket will be created
  + resource "aws_s3_bucket" "my_bucket" {
      + acceleration_status         = (known after apply)
      + acl                         = (known after apply)
      + arn                         = (known after apply)
      + bucket                      = "gabrielalobo"
      + bucket_domain_name          = (known after apply)
      + bucket_prefix               = (known after apply)
      + bucket_regional_domain_name = (known after apply)
      + force_destroy               = false
      + hosted_zone_id              = (known after apply)
      + id                          = (known after apply)
      + object_lock_enabled         = (known after apply)
      + policy                      = (known after apply)
      + region                      = (known after apply)
      + request_payer               = (known after apply)
      + tags                        = {
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + tags_all                    = {
          + "Environment" = "Test"
          + "Name"        = "MyS3Bucket"
        }
      + 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)
```



```
      + 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.

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_instance.my_ec2: Creating...
aws_s3_bucket.my_bucket: Still creating... [10s elapsed]
aws_instance.my_ec2: Still creating... [10s elapsed]
aws_s3_bucket.my_bucket: Still creating... [20s elapsed]
aws_instance.my_ec2: Still creating... [20s elapsed]
aws_s3_bucket.my_bucket: Creation complete after 20s [id=gabrielalobo]
aws_instance.my_ec2: Still creating... [30s elapsed]
aws_instance.my_ec2: Still creating... [40s elapsed]
aws_instance.my_ec2: Creation complete after 46s [id=i-027a5b622b95c2206]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

AWS



Recursos

Você está usando os seguintes recursos do Amazon EC2 na Região Leste dos EUA (Norte da Virgínia):

Instâncias (em execução)	1	Grupos de posicionamento	0	Grupos de segurança	1
Grupos do Auto Scaling	0	Hosts dedicados	0	Instâncias	1
IPs elásticos	0	Load balancers	0	Pares de chaves	0
Snapshots	0	Volumes	1		

Executar instância

Para começar, execute uma instância do Amazon EC2, que é um servidor virtual na nuvem.

Observação: suas instâncias serão executadas na Região Leste dos EUA (Norte da Virgínia)

Alarmes de instância

0 em alarme · 0 OK · 0 dados insuficientes

Integridade do serviço

Região: Leste dos EUA (Norte da Virgínia)

Status: Este serviço está funcionando normalmente

Zonas

Nome da zona	ID da zona
us-east-1a	use1-az4
us-east-1b	use1-az6

Nível gratuito do EC2 Informações

Ofertas para todas as regiões da AWS.

0 ofertas de nível gratuito do EC2 em uso

Previsão para o final do mês

0 ofertas previstas para exceder o limite do nível gratuito.

Excede o nível gratuito

0 ofertas foram excedidas e agora é preço conforme o uso.

Exibir os recursos globais do EC2

Exibir todas as ofertas de nível gratuito da AWS

Atributos da conta

VPC padrão

vpc-0ebdd2786d8a24c1b

Configurações

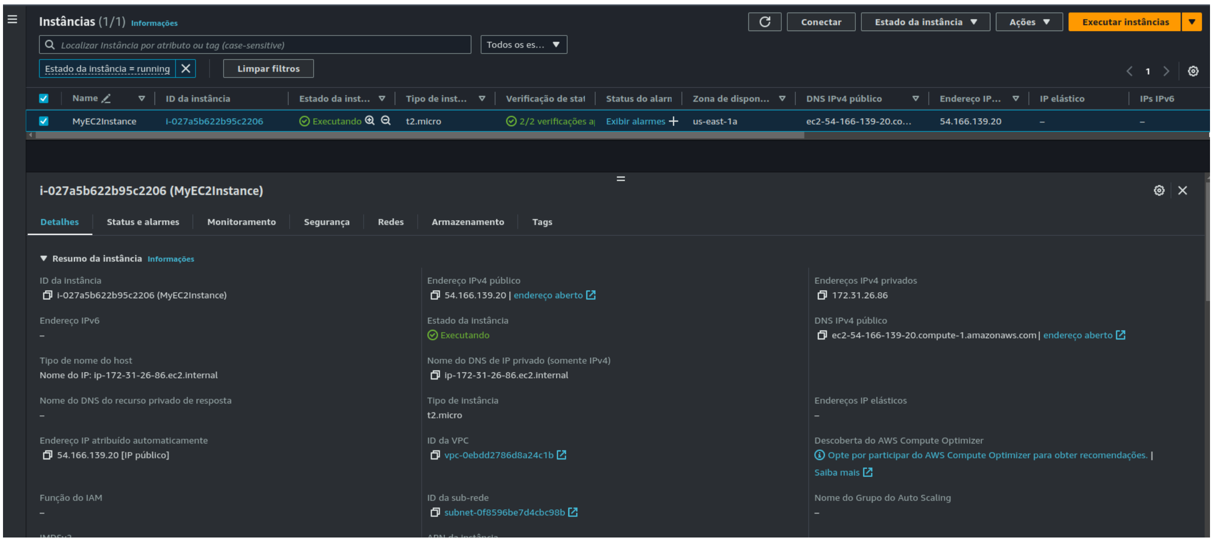
Proteção e segurança de dados

Zonas

Console serial do EC2

Especificação de crédito padrão

Preferências do console do EC2



Instâncias (1/1) Informações

Estado da instância = running

☑	Name	ID da instância	Estado da inst...	Tipo de inst...	Verificação de stat	Status do alarn	Zona de dispon...	DNS IPv4 público	Endereço IP...	IP elástico	IPs IPv6
☑	MyEC2Instance	i-027a5b622b95c2206	Executando	t2.micro	2/2 verificações a|	Exibir alarmes +	us-east-1a	ec2-54-166-139-20.co...	54.166.139.20	–	–

i-027a5b622b95c2206 (MyEC2Instance)

Resumo da instância Informações

ID da instância: i-027a5b622b95c2206 (MyEC2Instance)

Endereço IPv4 público: 54.166.139.20 | endereço aberto

Endereços IPv4 privados: 172.31.26.86

Endereço IPv6: –

Estado da instância: Executando

DNS IPv4 público: ec2-54-166-139-20.compute-1.amazonaws.com | endereço aberto

Tipo de nome do host: Nome do IP: ip-172-31-26-86.ec2.internal

Nome do DNS de IP privado (somente IPv4): ip-172-31-26-86.ec2.internal

Nome do DNS do recurso privado de resposta: –

Tipo de instância: t2.micro

Endereços IP elásticos: –

Endereço IP atribuído automaticamente: 54.166.139.20 [IP público]

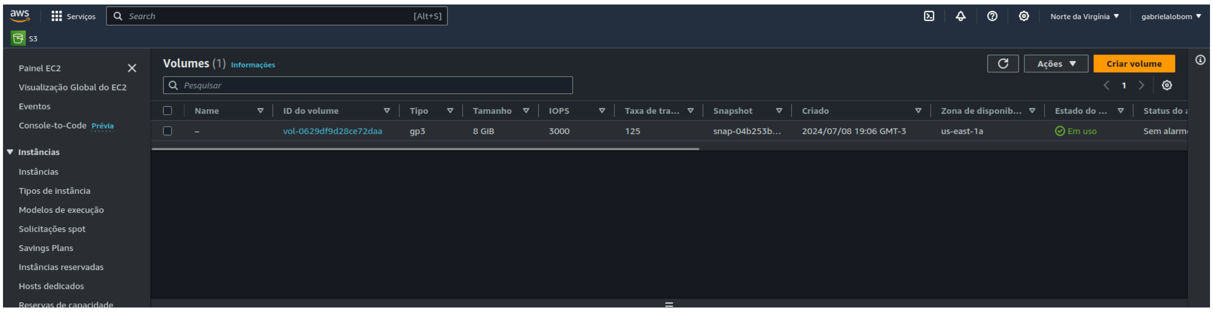
ID da VPC: vpc-0ebdd2786d8a24c1b

Descoberta do AWS Compute Optimizer: Opte por participar do AWS Compute Optimizer para obter recomendações. | Saiba mais

Função do IAM: –

ID da sub-rede: subnet-0f8596be7d4cbc98b

Nome do Grupo do Auto Scaling: –



Volumes (1) Informações

□	Name	ID do volume	Tipo	Tamanho	IOPS	Taxa de tra...	Snapshot	Criado	Zona de disponib...	Estado do ...	Status do a
□	–	vol-0629df9d28ce72daa	gp3	8 GiB	3000	125	snap-04b253b...	2024/07/08 19:06 GMT-3	us-east-1a	Em uso	Sem alarm