## S3 using Terraform

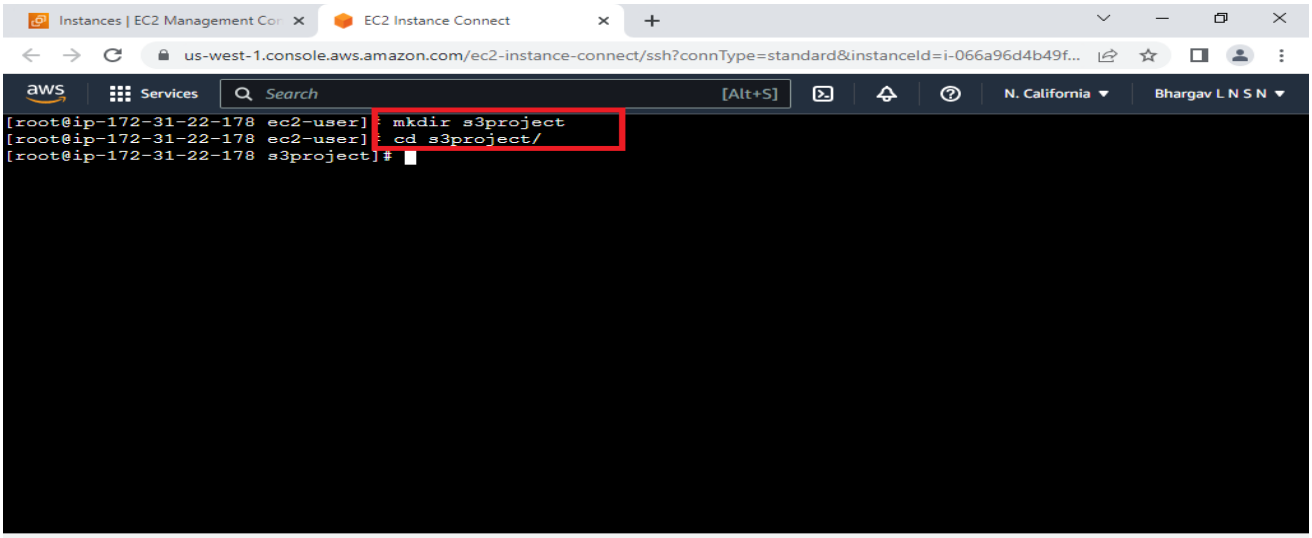
Step 1: Install terraform and open directory

Step2: create two files main.tf and variable.tf with below scripts.

```
resource "aws_s3_bucket" "my_bucket" {
  bucket = "terrafor-s3-bucket"
  acl = "private"
  versioning {
    enabled = "false"
  }
  force_destroy = "false"
}
variable "aws_access_key" {}
variable "aws_secret_key" {}
```

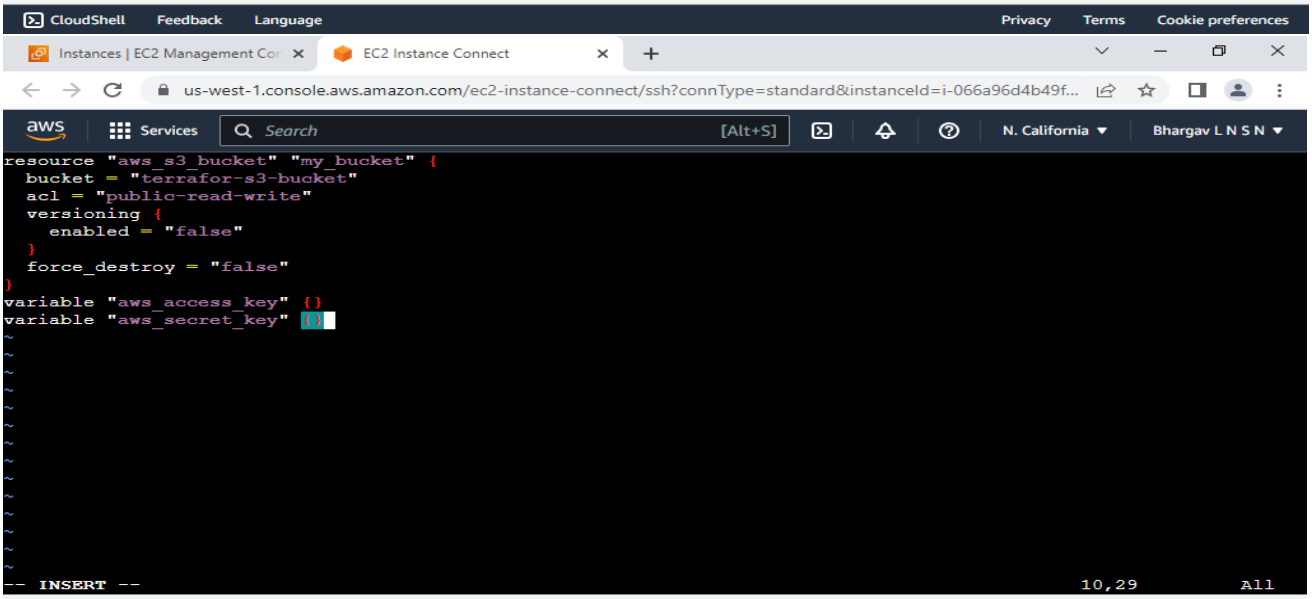
Provider.tf

```
provider "aws" {
access_key = "${var.aws_access_key}"
secret_key = "${var.aws_secret_key}"
region     = "us-west-1"
}
```



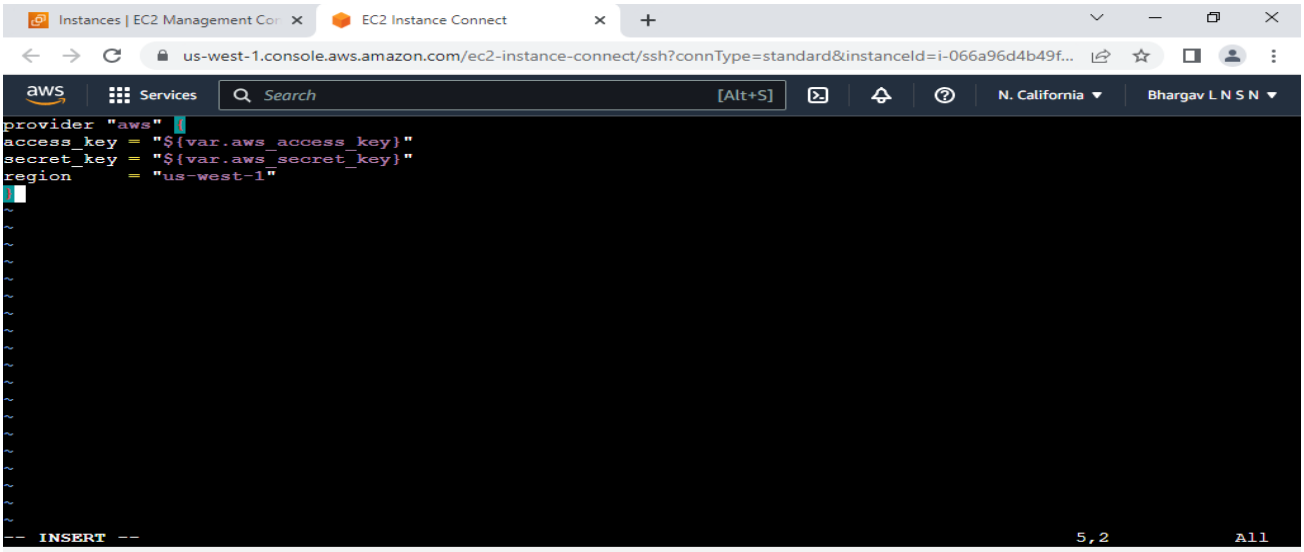

i-066a96d4b49f4915f (practice)

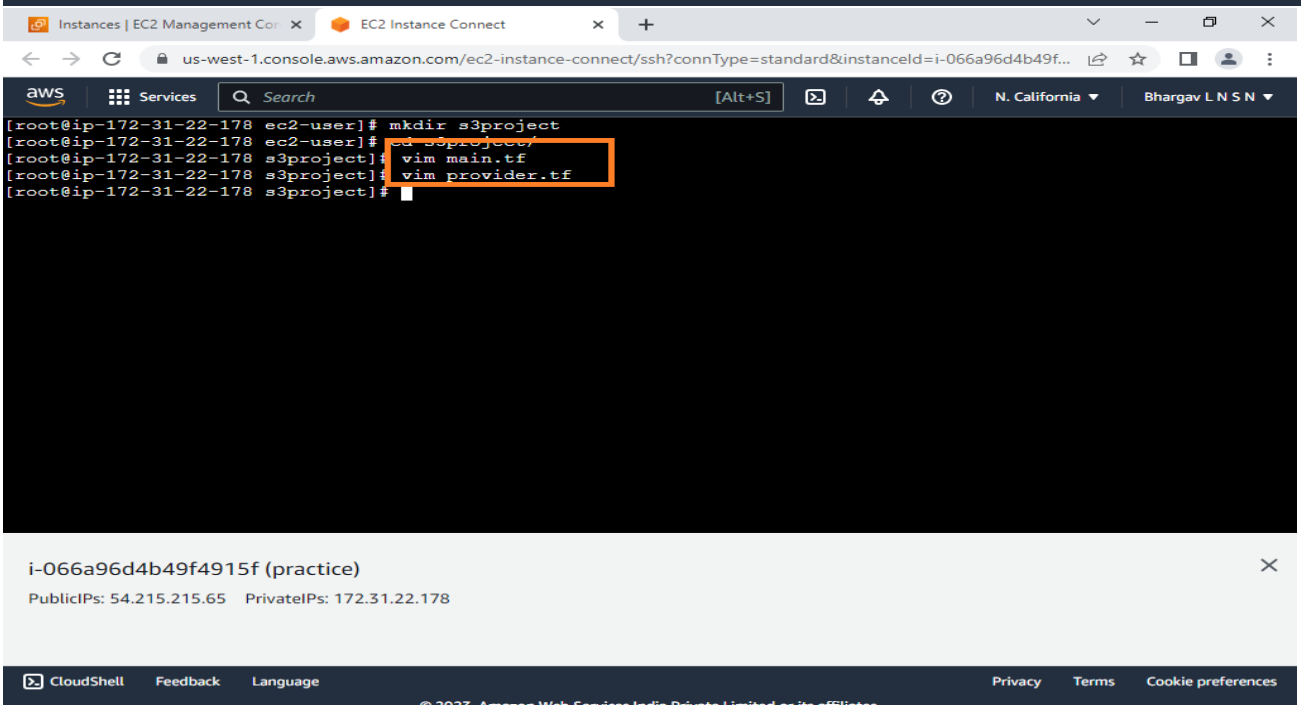
PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178

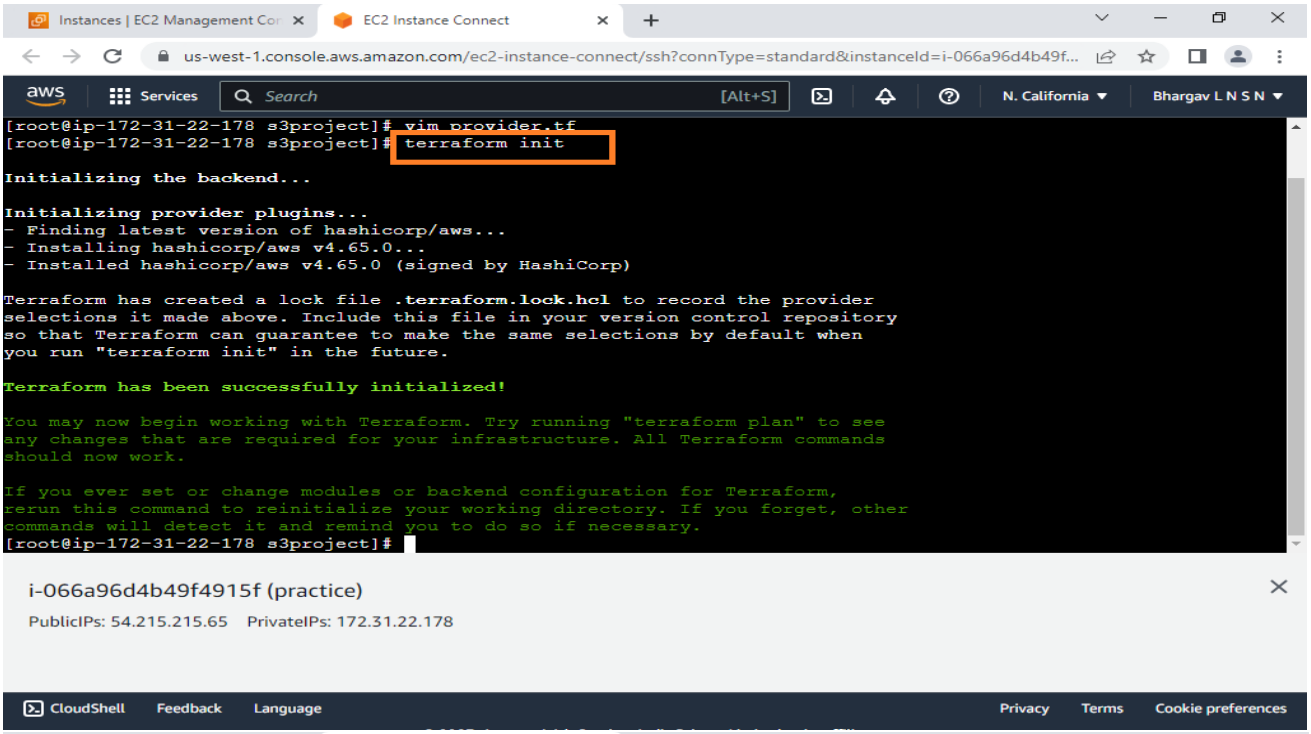


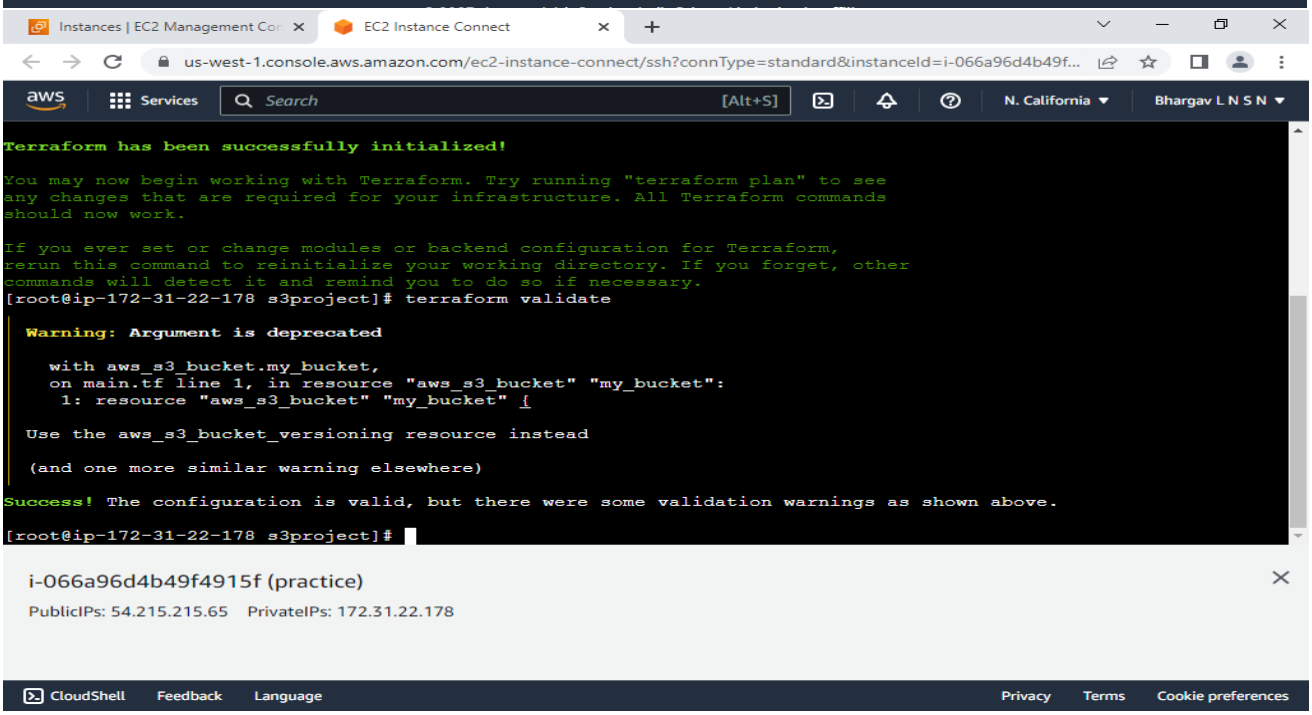

i-066a96d4b49f4915f (practice)

PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178


Instances | EC2 Management Con
EC2 Instance Connect
us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-066a96d4b49f...
aws
Services
Search
[Alt+S]
N. California
Bhargav L N S N
provider "aws" {
access_key = "${var.aws_access_key}"
secret_key = "${var.aws_secret_key}"
region = "us-west-1"
}
-- INSERT --
5,2
All
i-066a96d4b49f4915f (practice)
PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178
CloudShell
Feedback
Language
Privacy
Terms
Cookie preferences


Instances | EC2 Management Con
EC2 Instance Connect
us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-066a96d4b49f...
aws
Services
Search
[Alt+S]
N. California
Bhargav L N S N
[root@ip-172-31-22-178 ec2-user]# mkdir s3project
[root@ip-172-31-22-178 ec2-user]# cd s3project/
[root@ip-172-31-22-178 s3project]# vim main.tf
[root@ip-172-31-22-178 s3project]# vim provider.tf
[root@ip-172-31-22-178 s3project]#
i-066a96d4b49f4915f (practice)
PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178
CloudShell
Feedback
Language
Privacy
Terms
Cookie preferences


Instances | EC2 Management Cor
EC2 Instance Connect
us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-066a96d4b49f...
aws
Services
Search
[Alt+S]
N. California
Bhargav L N S N
[root@ip-172-31-22-178 s3project]# vim provider.tf
[root@ip-172-31-22-178 s3project]# terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v4.65.0...
- Installed hashicorp/aws v4.65.0 (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.
[root@ip-172-31-22-178 s3project]#
i-066a96d4b49f4915f (practice)
PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178
CloudShell
Feedback
Language
Privacy
Terms
Cookie preferences


Instances | EC2 Management Cor
EC2 Instance Connect
us-west-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-066a96d4b49f...
aws
Services
Search
[Alt+S]
N. California
Bhargav L N S N
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.
[root@ip-172-31-22-178 s3project]# terraform validate
Warning: Argument is deprecated
with aws_s3_bucket.my_bucket,
on main.tf line 1, in resource "aws_s3_bucket" "my_bucket":
1: resource "aws_s3_bucket" "my_bucket" {
Use the aws_s3_bucket_versioning resource instead
(and one more similar warning elsewhere)
Success! The configuration is valid, but there were some validation warnings as shown above.
[root@ip-172-31-22-178 s3project]#
i-066a96d4b49f4915f (practice)
PublicIPs: 54.215.215.65 PrivateIPs: 172.31.22.178
CloudShell
Feedback
Language
Privacy
Terms
Cookie preferences

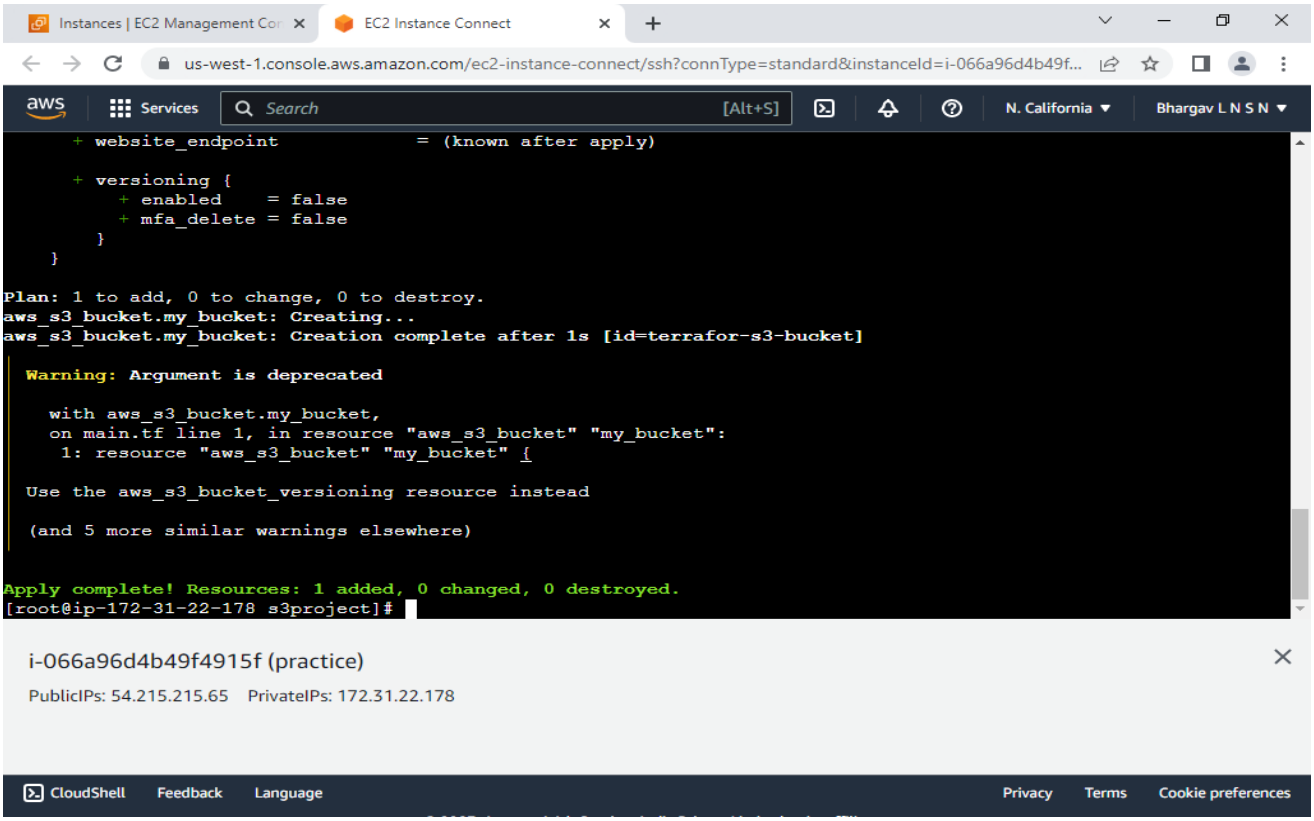
Step4: Now apply below commands to initialize terraform

i. Terraform init

ii. Terraform validate

iii. Terraform plan

iv. Terraform apply –auto-approve



S3 bucket has been created.