

# [Guvi Task]

## Day 1 - Iac - Terraform Introduction

**TASK:** Launch Linux EC2 instances in two regions using a single Terraform file.

The screenshot shows the AWS Cloud9 IDE interface. At the top, there's a navigation bar with 'aws' logo, 'Services', a search bar, and 'Mumbai' dropdown. Below the terminal window, a modal box displays information about an EC2 instance: 'i-0e764b4241da2399e (demo-terraform)', 'PublicIPs: 65.0.5.199', and 'PrivateIPs: 172.31.5.25'.

```
ubuntu@ip-172-31-5-25:~$ ls
demo.txt  ec2  localfile  samplefile
ubuntu@ip-172-31-5-25:~$ cd samplefile/
ubuntu@ip-172-31-5-25:~/samplefile$ vi sample.tf
ubuntu@ip-172-31-5-25:~/samplefile$ 
ubuntu@ip-172-31-5-25:~/samplefile$ 
```

The screenshot shows the AWS Cloud9 IDE interface. At the top, there's a navigation bar with 'aws' logo, 'Services', a search bar, and 'Mumbai' dropdown. Below the terminal window, a modal box displays information about an EC2 instance: 'i-0e764b4241da2399e (demo-terraform)', 'PublicIPs: 65.0.5.199', and 'PrivateIPs: 172.31.5.25'.

```
provider "aws" {
    alias = "ap_south_1"
    region = "ap-south-1"
}

provider "aws" {
    alias = "us_east_1"
    region = "us-east-1"
}

resource "aws_instance" "demo1" {
    provider = aws.ap_south_1

    ami = "ami-0ad21ae1d0696ad58"
    instance_type = "t2.micro"
    user_data = <<EOF
        #!/bin/bash
        echo "HELLO from ap-south-1" > /home/ubuntu/hello.txt
    EOF

    tags = {
        Name = "myec2withuserdata"
    }
}
-- INSERT --
```

AWS Services Search [Alt+S] Mumbai ▾ Sumit\_aws\_11 ▾

```
ami = "ami-0ad21ae1d0696ad58"
instance_type = "t2.micro"
user_data = <<EOF
#!/bin/bash
echo "HELLO from ap-south-1" > /home/ubuntu/hello.txt
EOF

tags = {
    Name = "myec2withuserdata"
}
}

resource "aws_instance" "demo2" {
provider = aws.us_east_2

ami = "ami-0649bea3443ede307"
instance_type = "t2.micro"
user_data = <<EOF
#!/bin/bash
echo "HELLO from us-east-2" > /home/ec2-user/hello.txt
EOF

tags = {
    Name = "myec2withuserdata"
}
}

-- INSERT --
```

32,75 86%

i-0e764b4241da2399e (demo-terraform) X

PublicIPs: 65.0.5.199 PrivateIPs: 172.31.5.25

AWS Services Search [Alt+S] Mumbai ▾ Sumit\_aws\_11 ▾

```
ubuntu@ip-172-31-5-25:~$ ls
demo.txt ec2 localfile samplefile/
ubuntu@ip-172-31-5-25:~$ cd samplefile/
ubuntu@ip-172-31-5-25:~/samplefile$ vi sample.tf
ubuntu@ip-172-31-5-25:~/samplefile$ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.57.0

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.
ubuntu@ip-172-31-5-25:~/samplefile$
```

i-0e764b4241da2399e (demo-terraform) X

PublicIPs: 65.0.5.199 PrivateIPs: 172.31.5.25

aws | Services | Search [Alt+S] | Mumbai | Sumit\_aws\_11

```
ubuntu@ip-172-31-5-25:~/samplefile$ terraform validate
Success! The configuration is valid.

ubuntu@ip-172-31-5-25:~/samplefile$ terraform plan
aws_instance.demo1: Refreshing state... [id=i-08087024e0077d7f3]

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.demo2 will be created
+ resource "aws_instance" "demo2" {
    + ami                               = "ami-0649bea3443ede307"
    + 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)

i-0e764b4241da2399e (demo-terraform)
```

PublicIPs: 65.0.5.199 PrivateIPs: 172.31.5.25

aws | Services | Search [Alt+S] | Mumbai | Sumit\_aws\_11

```
Plan: 1 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.

ubuntu@ip-172-31-5-25:~/samplefile$ terraform apply
aws_instance.demo1: Refreshing state... [id=i-08087024e0077d7f3]

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.demo2 will be created
+ resource "aws_instance" "demo2" {
    + ami                               = "ami-0649bea3443ede307"
    + 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)

i-0e764b4241da2399e (demo-terraform)
```

DiskUsed 65.0 E 100 | DrivedUsed 172.31 E 25

aws | Services | Search [Alt+S] | Mumbai | Sumit\_aws\_11

```

+ metadata_options (known after apply)
+ network_interface (known after apply)
+ private_dns_name_options (known after apply)
+ root_block_device (known after apply)
}

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 instance.demo2: Creating...
aws_instance.demo2: Still creating... [10s elapsed]
aws_instance.demo2: Still creating... [20s elapsed]
aws_instance.demo2: Still creating... [30s elapsed]
aws_instance.demo2: Creation complete after 35s [id=i-0cb014d0891d32b12]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-5-25:~/samplefile$ 

```

i-0e764b4241da2399e (demo-terraform)  
PublicIPs: 65.0.5.199 PrivateIPs: 172.31.5.25

aws | Services | Search [Alt+S] | Mumbai | Sumit-dhal @ 7303-3534-6690

**Instances (1/3) Info**

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
demo-terraform	i-0e764b4241da2399e	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1
myterraform-i...	i-07328f40f9081b4f5	Running	t2.micro	2/2 checks passed	View alarms +	ap-south-1
<b>myec2withuse...</b>	<b>i-08087024e0077d7f3</b>	<b>Running</b>	<b>t2.micro</b>	<b>2/2 checks passed</b>	<b>View alarms +</b>	<b>ap-south-1</b>

**i-08087024e0077d7f3 (myec2withuserdata)**

**Instance summary**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-08087024e0077d7f3 (myec2withuserdata)	43.204.108.151   open address	172.31.3.10
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-43-204-108-151.ap-south-1.compute.amazonaws.com   open address
Hostname type	Private IP DNS name (IPv4 only)	

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

AWS Services Search [Alt+S] Mumbai Sumit-dhal @ sumithdal

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.  
See <https://ubuntu.com/esm> or run: sudo pro status

The list of available updates is more than a week old.  
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/\*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo\_root" for details.

```
ubuntu@ip-172-31-3-10:~$ ls
hello.txt
ubuntu@ip-172-31-3-10:~$
```

i-08087024e0077d7f3 (myec2withuserdata)

PublicIPs: 43.204.108.151 PrivateIPs: 172.31.3.10

AWS Services Search [Alt+S] Ohio Sumit-dhal @ 7303-3534-6690

**Instances (1/1) Info**

Find Instance by attribute or tag (case-sensitive) All states

Instance state = running Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability zone
myec2withuse...	i-0cb014d0891d32b12	Running	t2.micro	2/2 checks passed	View alarms +	us-east-2b

**i-0cb014d0891d32b12 (myec2withuserdata)**

Details Status and alarms Monitoring Security Networking Storage Tags

**Instance summary Info**

Instance ID	Public IPv4 address	Private IPv4 addresses
i-0cb014d0891d32b12 (myec2withuserdata)	52.14.200.191   open address	172.31.31.53
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-52-14-200-191.us-east-2.compute.amazonaws.com   open address

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows a CloudShell terminal window. At the top, there's a navigation bar with the AWS logo, 'Services' (represented by a grid icon), a search bar containing 'Search [Alt+S]', and user information 'Ohio | Sumit-dhal @ sumitdhal'. Below the bar, a large portion of the screen is a terminal window. Inside the terminal, there's a multi-line ASCII art logo for Amazon Linux 2023, followed by the command 'ls' which outputs 'hello.txt'. At the bottom of the terminal window, it says 'PublicIPs: 52.14.200.191 PrivateIPs: 172.31.31.53'. A small 'X' icon is located in the top right corner of the terminal area.

```
'~\_\#\#\#_      Amazon Linux 2023
~~ \_\#\#\#\\
~~ \|##|
~~ \#/| https://aws.amazon.com/linux/amazon-linux-2023
~~   V~'__->
~~   /
~~_-'_/
~/m'_
[ec2-user@ip-172-31-31-53 ~]$ ls
hello.txt
[ec2-user@ip-172-31-31-53 ~]$ 
```

i-0cb014d0891d32b12 (myec2withuserdata) X

PublicIPs: 52.14.200.191 PrivateIPs: 172.31.31.53