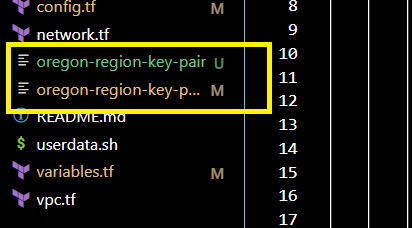
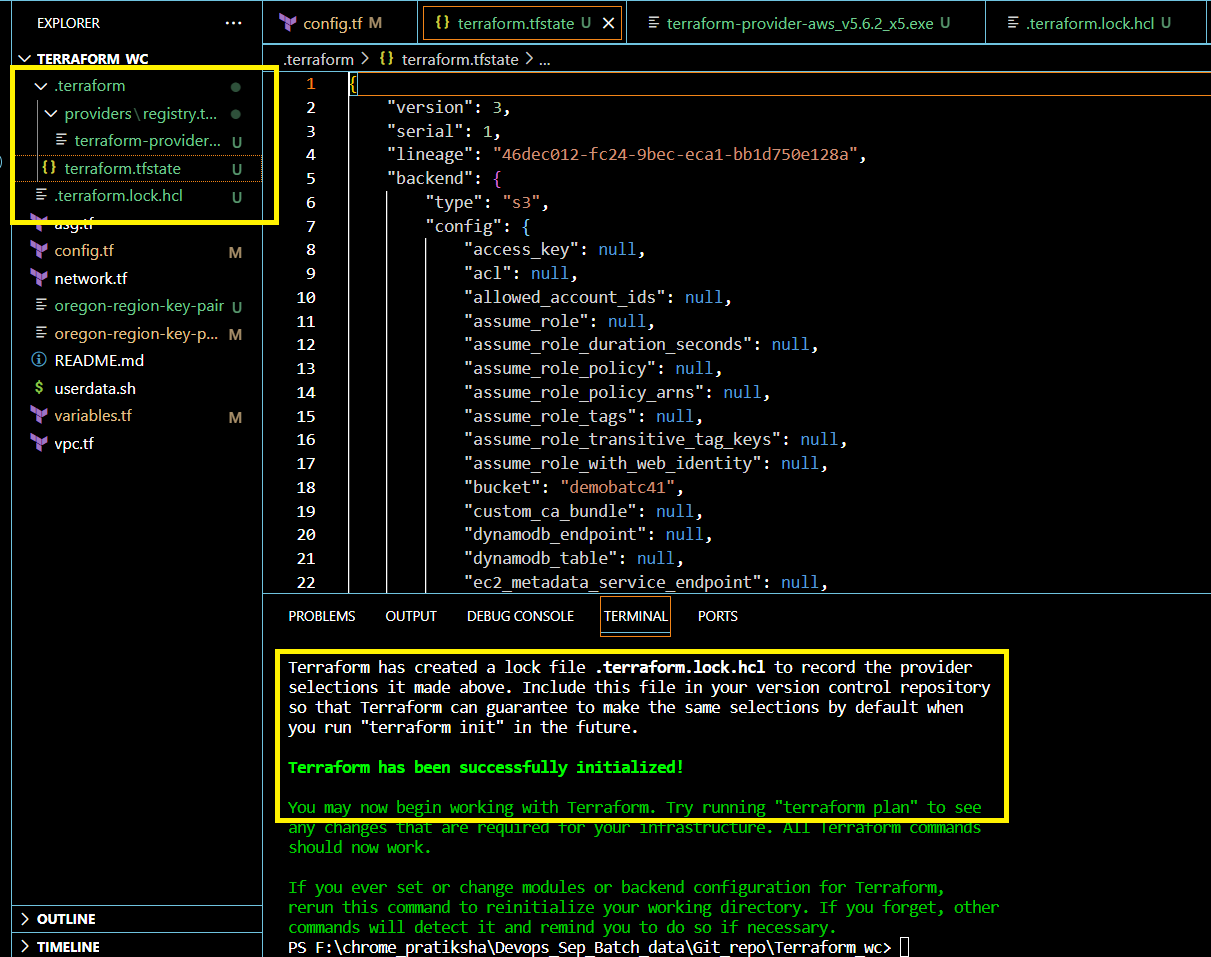
Terraform Output:

Cmd 1: ssh-keygen -f oregon-region-key-pair



Cmd 2: terraform init



\

Cmd 3: terraform plan -out "file.plan"

---------Output got on cmd window

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\_autoscaling\_group.dev-autoscaling-group will be created

+ resource "aws\_autoscaling\_group" "dev-autoscaling-group" {

+ arn = (known after apply)

+ availability\_zones = (known after apply)

+ default\_cooldown = (known after apply)

+ desired\_capacity = (known after apply)

+ force\_delete = false

+ force\_delete\_warm\_pool = false

+ health\_check\_grace\_period = 300

+ health\_check\_type = (known after apply)

+ id = (known after apply)

+ launch\_configuration = "dev-launch-config"

+ load\_balancers = (known after apply)

+ max\_size = 1

+ metrics\_granularity = "1Minute"

+ min\_size = 1

+ name = "dev-asg"

+ name\_prefix = (known after apply)

+ predicted\_capacity = (known after apply)

+ protect\_from\_scale\_in = false

+ service\_linked\_role\_arn = (known after apply)

+ target\_group\_arns = (known after apply)

+ vpc\_zone\_identifier = (known after apply)

+ wait\_for\_capacity\_timeout = "10m"

+ warm\_pool\_size = (known after apply)

+ tag {

+ key = "Name"

+ propagate\_at\_launch = true

+ value = "dev-test"

}

}

# aws\_internet\_gateway.dev-igw will be created

+ resource "aws\_internet\_gateway" "dev-igw" {

+ arn = (known after apply)

+ id = (known after apply)

+ owner\_id = (known after apply)

+ tags = {

+ "Name" = "dev-igw"

}

+ tags\_all = {

+ "Name" = "dev-igw"

}

+ vpc\_id = (known after apply)

}

# aws\_key\_pair.oregon-region-key-pair will be created

+ resource "aws\_key\_pair" "oregon-region-key-pair" {

+ arn = (known after apply)

+ fingerprint = (known after apply)

+ id = (known after apply)

+ key\_name = "oregon-region-key-pair"

+ key\_name\_prefix = (known after apply)

+ key\_pair\_id = (known after apply)

+ key\_type = (known after apply)

+ public\_key = "ssh-rsa  admin@DESKTOP-3R484SM"

+ tags\_all = (known after apply)

}

# aws\_launch\_configuration.dev-launch-config will be created

+ resource "aws\_launch\_configuration" "dev-launch-config" {

+ arn = (known after apply)

+ associate\_public\_ip\_address = true

+ ebs\_optimized = (known after apply)

+ enable\_monitoring = true

+ id = (known after apply)

+ image\_id = "ami-0d593311db5abb72b"

+ instance\_type = "t2.micro"

+ key\_name = (known after apply)

+ name = "dev-launch-config"

+ name\_prefix = (known after apply)

+ security\_groups = (known after apply)

+ user\_data = "f39b9fb514e7a28be22470e95646b1d9ba99c867"

}

# aws\_route\_table.dev-public-crt will be created

+ resource "aws\_route\_table" "dev-public-crt" {

+ arn = (known after apply)

+ id = (known after apply)

+ owner\_id = (known after apply)

+ propagating\_vgws = (known after apply)

+ route = [

+ {

+ carrier\_gateway\_id = ""

+ cidr\_block = "0.0.0.0/0"

+ core\_network\_arn = ""

+ destination\_prefix\_list\_id = ""

+ egress\_only\_gateway\_id = ""

+ gateway\_id = (known after apply)

+ ipv6\_cidr\_block = ""

+ local\_gateway\_id = ""

+ nat\_gateway\_id = ""

+ network\_interface\_id = ""

+ transit\_gateway\_id = ""

+ vpc\_endpoint\_id = ""

+ vpc\_peering\_connection\_id = ""

},

]

+ tags = {

+ "Name" = "dev-public-crt"

}

+ tags\_all = {

+ "Name" = "dev-public-crt"

}

+ vpc\_id = (known after apply)

}

# aws\_route\_table\_association.dev-crta-public-subnet-1 will be created

+ resource "aws\_route\_table\_association" "dev-crta-public-subnet-1" {

+ id = (known after apply)

+ route\_table\_id = (known after apply)

+ subnet\_id = (known after apply)

}

# aws\_security\_group.ssh-allowed will be created

+ resource "aws\_security\_group" "ssh-allowed" {

+ arn = (known after apply)

+ description = "Managed by Terraform"

+ egress = [

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 0

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "-1"

+ security\_groups = []

+ self = false

+ to\_port = 0

},

]

+ id = (known after apply)

+ ingress = [

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 22

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "tcp"

+ security\_groups = []

+ self = false

+ to\_port = 22

},

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 80

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "tcp"

+ security\_groups = []

+ self = false

+ to\_port = 80

},

]

+ name = (known after apply)

+ name\_prefix = (known after apply)

+ owner\_id = (known after apply)

+ revoke\_rules\_on\_delete = false

+ tags = {

+ "Name" = "ssh-allowed"

}

+ tags\_all = {

+ "Name" = "ssh-allowed"

}

+ vpc\_id = (known after apply)

}

# aws\_subnet.dev-subnet-public-1 will be created

+ resource "aws\_subnet" "dev-subnet-public-1" {

+ arn = (known after apply)

+ assign\_ipv6\_address\_on\_creation = false

+ availability\_zone = "us-west-2a"

+ availability\_zone\_id = (known after apply)

+ cidr\_block = "10.0.1.0/24"

+ enable\_dns64 = false

+ enable\_resource\_name\_dns\_a\_record\_on\_launch = false

+ enable\_resource\_name\_dns\_aaaa\_record\_on\_launch = false

+ id = (known after apply)

+ ipv6\_cidr\_block\_association\_id = (known after apply)

+ ipv6\_native = false

+ map\_public\_ip\_on\_launch = true

+ owner\_id = (known after apply)

+ private\_dns\_hostname\_type\_on\_launch = (known after apply)

+ tags = {

+ "Name" = "dev-subnet-public-1"

}

+ tags\_all = {

+ "Name" = "dev-subnet-public-1"

}

+ vpc\_id = (known after apply)

}

# aws\_vpc.dev-vpc will be created

+ resource "aws\_vpc" "dev-vpc" {

+ arn = (known after apply)

+ cidr\_block = "10.0.0.0/16"

+ default\_network\_acl\_id = (known after apply)

+ default\_route\_table\_id = (known after apply)

+ default\_security\_group\_id = (known after apply)

+ dhcp\_options\_id = (known after apply)

+ enable\_dns\_hostnames = true

+ enable\_dns\_support = true

+ enable\_network\_address\_usage\_metrics = (known after apply)

+ id = (known after apply)

+ instance\_tenancy = "default"

+ ipv6\_association\_id = (known after apply)

+ ipv6\_cidr\_block = (known after apply)

+ ipv6\_cidr\_block\_network\_border\_group = (known after apply)

+ main\_route\_table\_id = (known after apply)

+ owner\_id = (known after apply)

+ tags = {

+ "Name" = "dev-vpc"

}

+ tags\_all = {

+ "Name" = "dev-vpc"

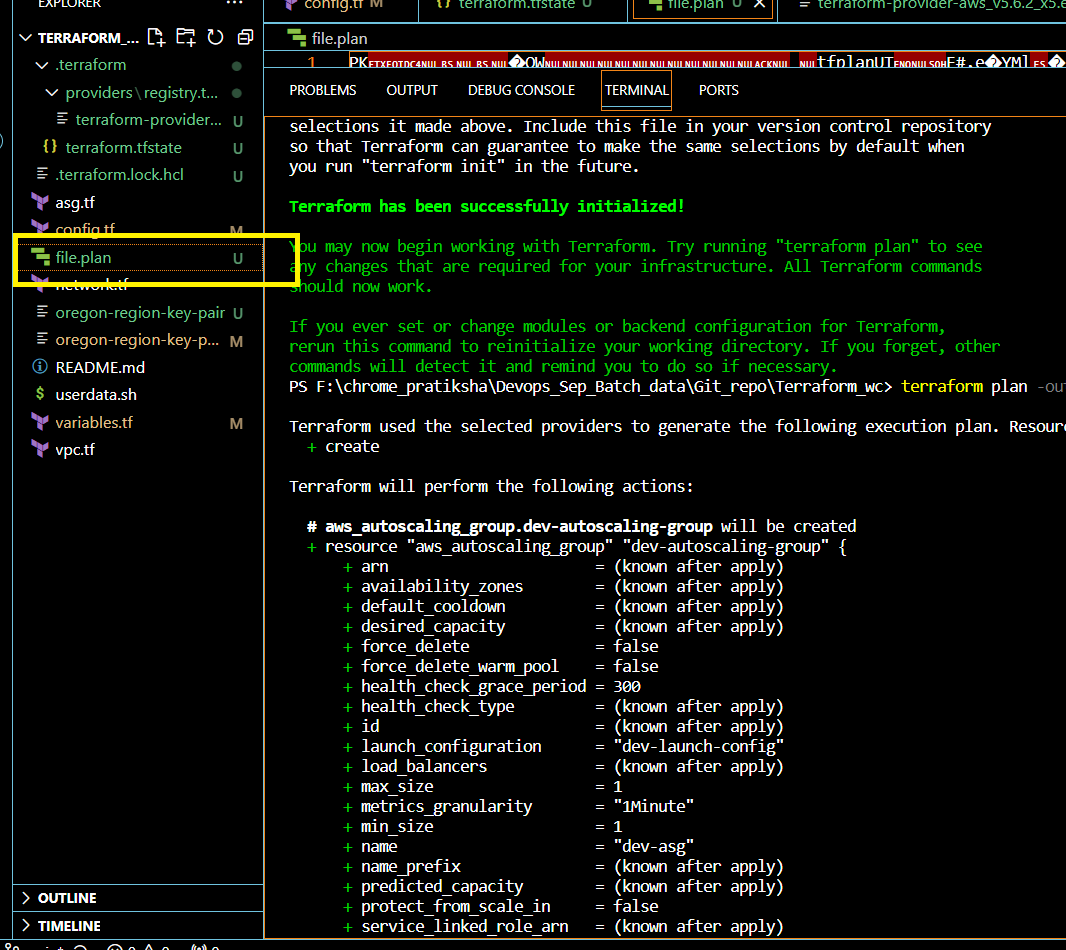
}

}

Plan: 9 to add, 0 to change, 0 to destroy.

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

Saved the plan to: file.plan



Cmd 4: terraform apply

{Note: As e just passed apply cmd it had first created plan file and then applied it if you want just apply created plan file then use: terraform apply “file1.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\_autoscaling\_group.dev-autoscaling-group will be created

+ resource "aws\_autoscaling\_group" "dev-autoscaling-group" {

+ arn = (known after apply)

+ availability\_zones = (known after apply)

+ default\_cooldown = (known after apply)

+ desired\_capacity = (known after apply)

+ force\_delete = false

+ force\_delete\_warm\_pool = false

+ health\_check\_grace\_period = 300

+ health\_check\_type = (known after apply)

+ id = (known after apply)

+ launch\_configuration = "dev-launch-config"

+ load\_balancers = (known after apply)

+ max\_size = 1

+ metrics\_granularity = "1Minute"

+ min\_size = 1

+ name = "dev-asg"

+ name\_prefix = (known after apply)

+ predicted\_capacity = (known after apply)

+ protect\_from\_scale\_in = false

+ service\_linked\_role\_arn = (known after apply)

+ target\_group\_arns = (known after apply)

+ vpc\_zone\_identifier = (known after apply)

+ wait\_for\_capacity\_timeout = "10m"

+ warm\_pool\_size = (known after apply)

+ tag {

+ key = "Name"

+ propagate\_at\_launch = true

+ value = "dev-test"

}

}

# aws\_internet\_gateway.dev-igw will be created

+ resource "aws\_internet\_gateway" "dev-igw" {

+ arn = (known after apply)

+ id = (known after apply)

+ owner\_id = (known after apply)

+ tags = {

+ "Name" = "dev-igw"

}

+ tags\_all = {

+ "Name" = "dev-igw"

}

+ vpc\_id = (known after apply)

}

# aws\_key\_pair.oregon-region-key-pair will be created

+ resource "aws\_key\_pair" "oregon-region-key-pair" {

+ arn = (known after apply)

+ fingerprint = (known after apply)

+ id = (known after apply)

+ key\_name = "oregon-region-key-pair"

+ key\_name\_prefix = (known after apply)

+ key\_pair\_id = (known after apply)

+ key\_type = (known after apply)

+ public\_key = "ssh-rsa  admin@DESKTOP-3R484SM"

+ tags\_all = (known after apply)

}

# aws\_launch\_configuration.dev-launch-config will be created

+ resource "aws\_launch\_configuration" "dev-launch-config" {

+ arn = (known after apply)

+ associate\_public\_ip\_address = true

+ ebs\_optimized = (known after apply)

+ enable\_monitoring = true

+ id = (known after apply)

+ image\_id = "ami-0d593311db5abb72b"

+ instance\_type = "t2.micro"

+ key\_name = (known after apply)

+ name = "dev-launch-config"

+ name\_prefix = (known after apply)

+ security\_groups = (known after apply)

+ user\_data = "f39b9fb514e7a28be22470e95646b1d9ba99c867"

}

# aws\_route\_table.dev-public-crt will be created

+ resource "aws\_route\_table" "dev-public-crt" {

+ arn = (known after apply)

+ id = (known after apply)

+ owner\_id = (known after apply)

+ propagating\_vgws = (known after apply)

+ route = [

+ {

+ carrier\_gateway\_id = ""

+ cidr\_block = "0.0.0.0/0"

+ core\_network\_arn = ""

+ destination\_prefix\_list\_id = ""

+ egress\_only\_gateway\_id = ""

+ gateway\_id = (known after apply)

+ ipv6\_cidr\_block = ""

+ local\_gateway\_id = ""

+ nat\_gateway\_id = ""

+ network\_interface\_id = ""

+ transit\_gateway\_id = ""

+ vpc\_endpoint\_id = ""

+ vpc\_peering\_connection\_id = ""

},

]

+ tags = {

+ "Name" = "dev-public-crt"

}

+ tags\_all = {

+ "Name" = "dev-public-crt"

}

+ vpc\_id = (known after apply)

}

# aws\_route\_table\_association.dev-crta-public-subnet-1 will be created

+ resource "aws\_route\_table\_association" "dev-crta-public-subnet-1" {

+ id = (known after apply)

+ route\_table\_id = (known after apply)

+ subnet\_id = (known after apply)

}

# aws\_security\_group.ssh-allowed will be created

+ resource "aws\_security\_group" "ssh-allowed" {

+ arn = (known after apply)

+ description = "Managed by Terraform"

+ egress = [

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 0

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "-1"

+ security\_groups = []

+ self = false

+ to\_port = 0

},

]

+ id = (known after apply)

+ ingress = [

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 22

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "tcp"

+ security\_groups = []

+ self = false

+ to\_port = 22

},

+ {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ description = ""

+ from\_port = 80

+ ipv6\_cidr\_blocks = []

+ prefix\_list\_ids = []

+ protocol = "tcp"

+ security\_groups = []

+ self = false

+ to\_port = 80

},

]

+ name = (known after apply)

+ name\_prefix = (known after apply)

+ owner\_id = (known after apply)

+ revoke\_rules\_on\_delete = false

+ tags = {

+ "Name" = "ssh-allowed"

}

+ tags\_all = {

+ "Name" = "ssh-allowed"

}

+ vpc\_id = (known after apply)

}

# aws\_subnet.dev-subnet-public-1 will be created

+ resource "aws\_subnet" "dev-subnet-public-1" {

+ arn = (known after apply)

+ assign\_ipv6\_address\_on\_creation = false

+ availability\_zone = "us-west-2a"

+ availability\_zone\_id = (known after apply)

+ cidr\_block = "10.0.1.0/24"

+ enable\_dns64 = false

+ enable\_resource\_name\_dns\_a\_record\_on\_launch = false

+ enable\_resource\_name\_dns\_aaaa\_record\_on\_launch = false

+ id = (known after apply)

+ ipv6\_cidr\_block\_association\_id = (known after apply)

+ ipv6\_native = false

+ map\_public\_ip\_on\_launch = true

+ owner\_id = (known after apply)

+ private\_dns\_hostname\_type\_on\_launch = (known after apply)

+ tags = {

+ "Name" = "dev-subnet-public-1"

}

+ tags\_all = {

+ "Name" = "dev-subnet-public-1"

}

+ vpc\_id = (known after apply)

}

# aws\_vpc.dev-vpc will be created

+ resource "aws\_vpc" "dev-vpc" {

+ arn = (known after apply)

+ cidr\_block = "10.0.0.0/16"

+ default\_network\_acl\_id = (known after apply)

+ default\_route\_table\_id = (known after apply)

+ default\_security\_group\_id = (known after apply)

+ dhcp\_options\_id = (known after apply)

+ enable\_dns\_hostnames = true

+ enable\_dns\_support = true

+ enable\_network\_address\_usage\_metrics = (known after apply)

+ id = (known after apply)

+ instance\_tenancy = "default"

+ ipv6\_association\_id = (known after apply)

+ ipv6\_cidr\_block = (known after apply)

+ ipv6\_cidr\_block\_network\_border\_group = (known after apply)

+ main\_route\_table\_id = (known after apply)

+ owner\_id = (known after apply)

+ tags = {

+ "Name" = "dev-vpc"

}

+ tags\_all = {

+ "Name" = "dev-vpc"

}

}

Plan: 9 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\_vpc.dev-vpc: Creating...

aws\_key\_pair.oregon-region-key-pair: Creating...

aws\_key\_pair.oregon-region-key-pair: Creation complete after 2s [id=oregon-region-key-pair]

aws\_vpc.dev-vpc: Still creating... [10s elapsed]

aws\_vpc.dev-vpc: Creation complete after 15s [id=vpc-03eaf9bbde4f1e8b3]

aws\_internet\_gateway.dev-igw: Creating...

aws\_subnet.dev-subnet-public-1: Creating...

aws\_security\_group.ssh-allowed: Creating...

aws\_internet\_gateway.dev-igw: Creation complete after 2s [id=igw-0d4e51cf8d7197cea]

aws\_route\_table.dev-public-crt: Creating...

aws\_route\_table.dev-public-crt: Creation complete after 3s [id=rtb-0070d25b1f367b7ed]

aws\_security\_group.ssh-allowed: Creation complete after 6s [id=sg-008ab9c07c388b518]

aws\_launch\_configuration.dev-launch-config: Creating...

aws\_launch\_configuration.dev-launch-config: Creation complete after 2s [id=dev-launch-config]

aws\_subnet.dev-subnet-public-1: Still creating... [10s elapsed]

aws\_subnet.dev-subnet-public-1: Creation complete after 13s [id=subnet-0494d14e613cf2bb0]

aws\_route\_table\_association.dev-crta-public-subnet-1: Creating...

aws\_autoscaling\_group.dev-autoscaling-group: Creating...

aws\_route\_table\_association.dev-crta-public-subnet-1: Creation complete after 1s [id=rtbassoc-03ce78393acbf1397]

aws\_autoscaling\_group.dev-autoscaling-group: Still creating... [10s elapsed]

aws\_autoscaling\_group.dev-autoscaling-group: Still creating... [20s elapsed]

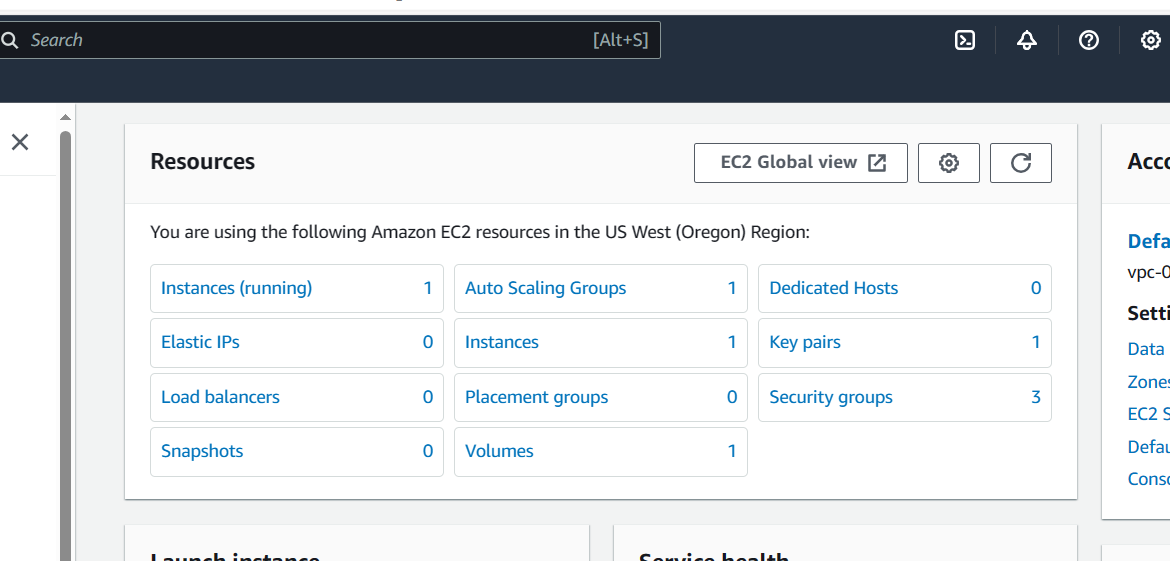
aws\_autoscaling\_group.dev-autoscaling-group: Still creating... [30s elapsed]

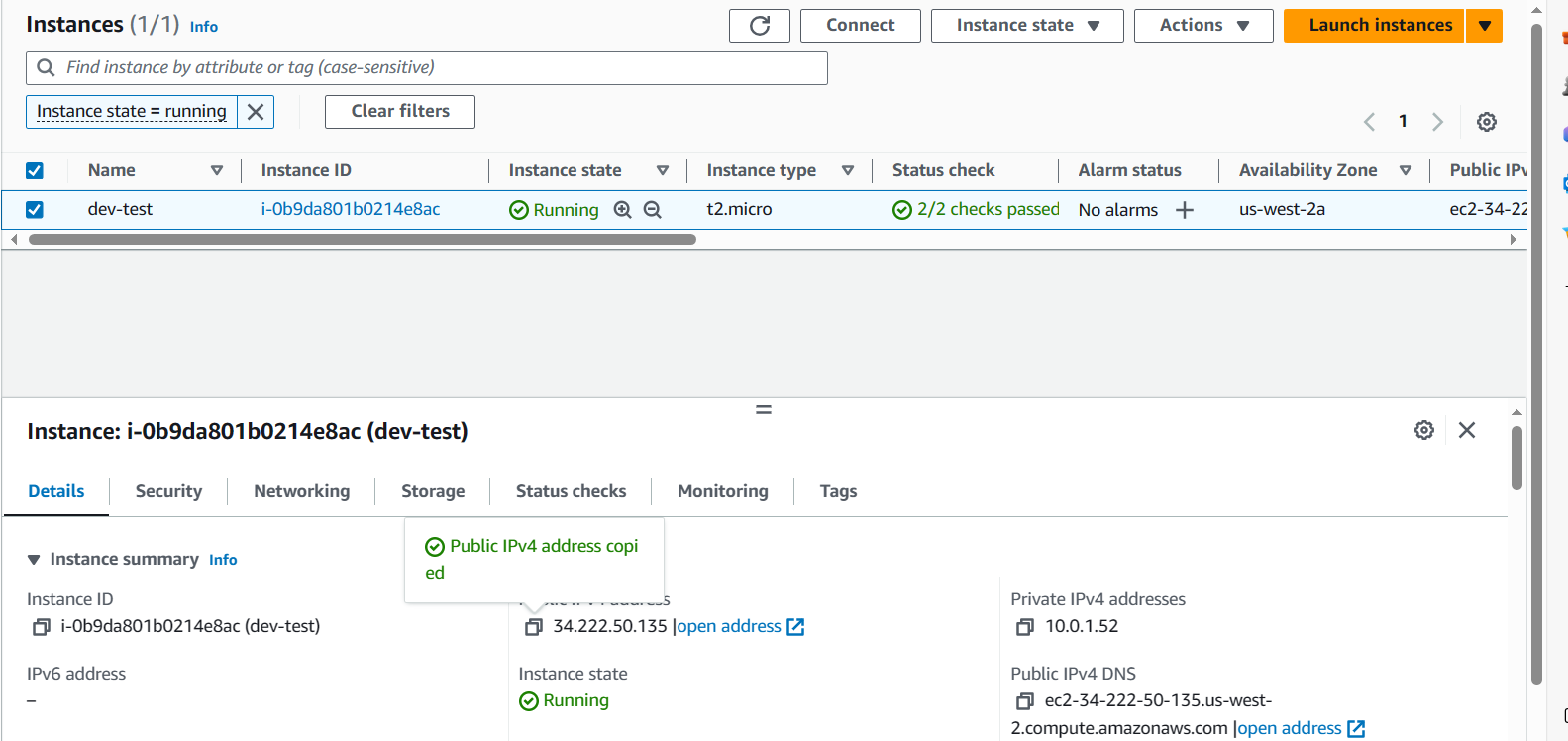
aws\_autoscaling\_group.dev-autoscaling-group: Still creating... [40s elapsed]

aws\_autoscaling\_group.dev-autoscaling-group: Creation complete after 44s [id=dev-asg]

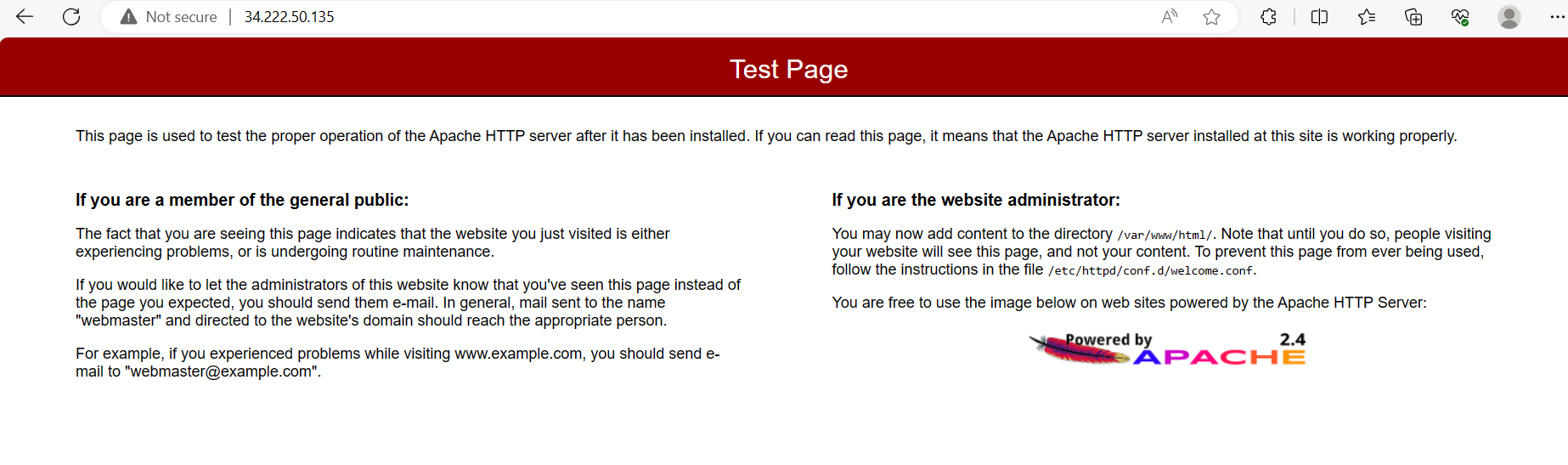
Apply complete! Resources: 9 added, 0 changed, 0 destroyed.

1. Created EC2 instance

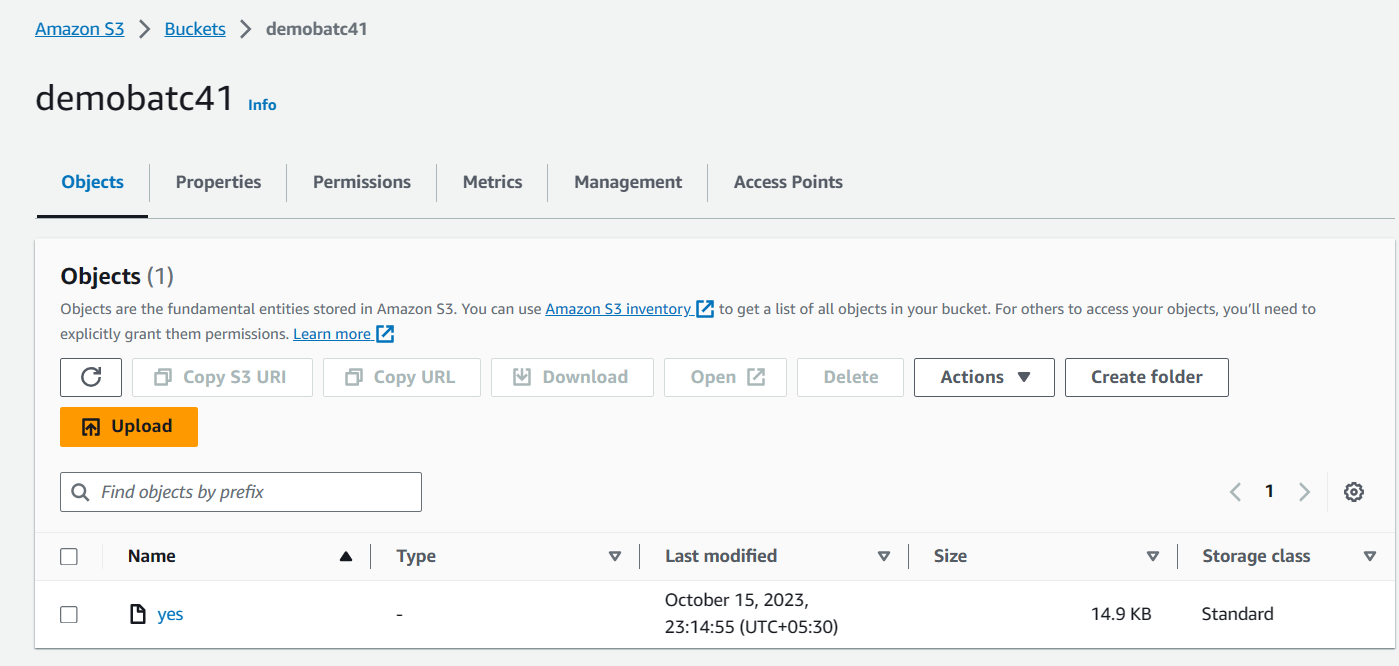




1. As http apache is also installed so working page with ip



1. Created yes named state file in s3 bucket



File :

