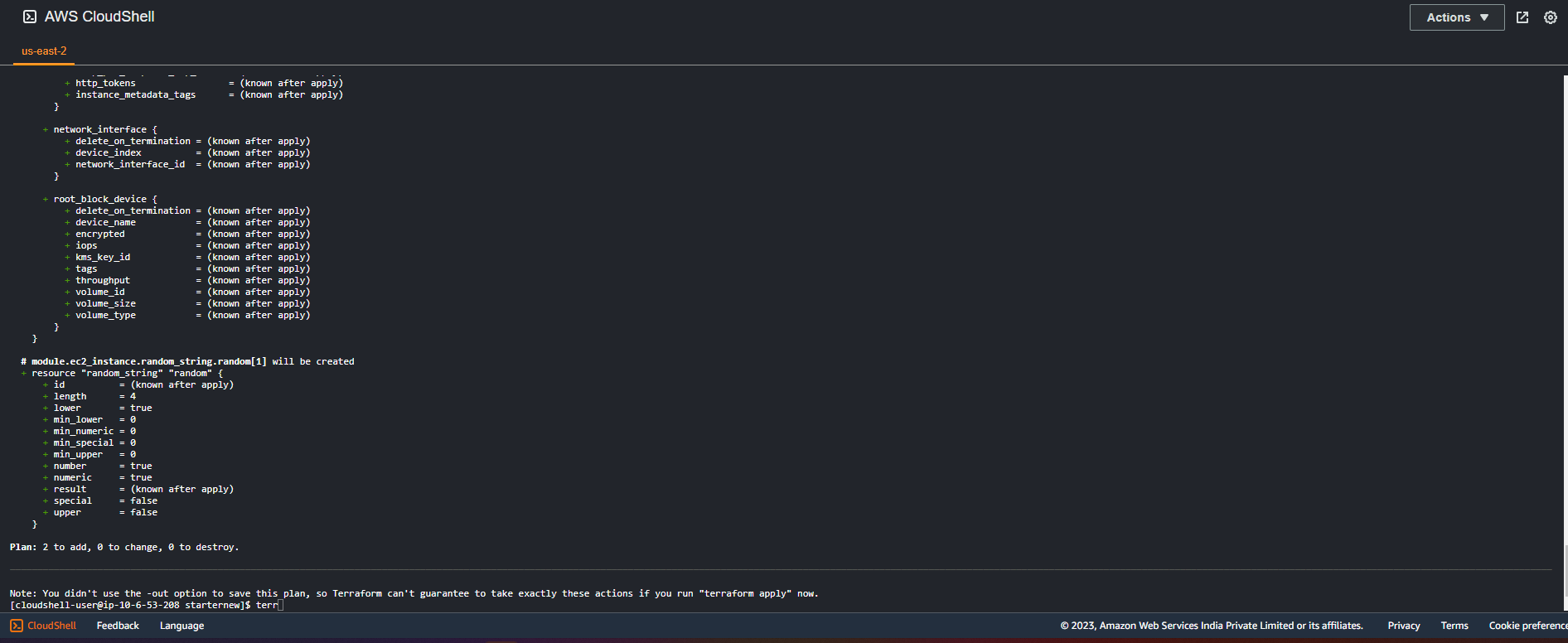
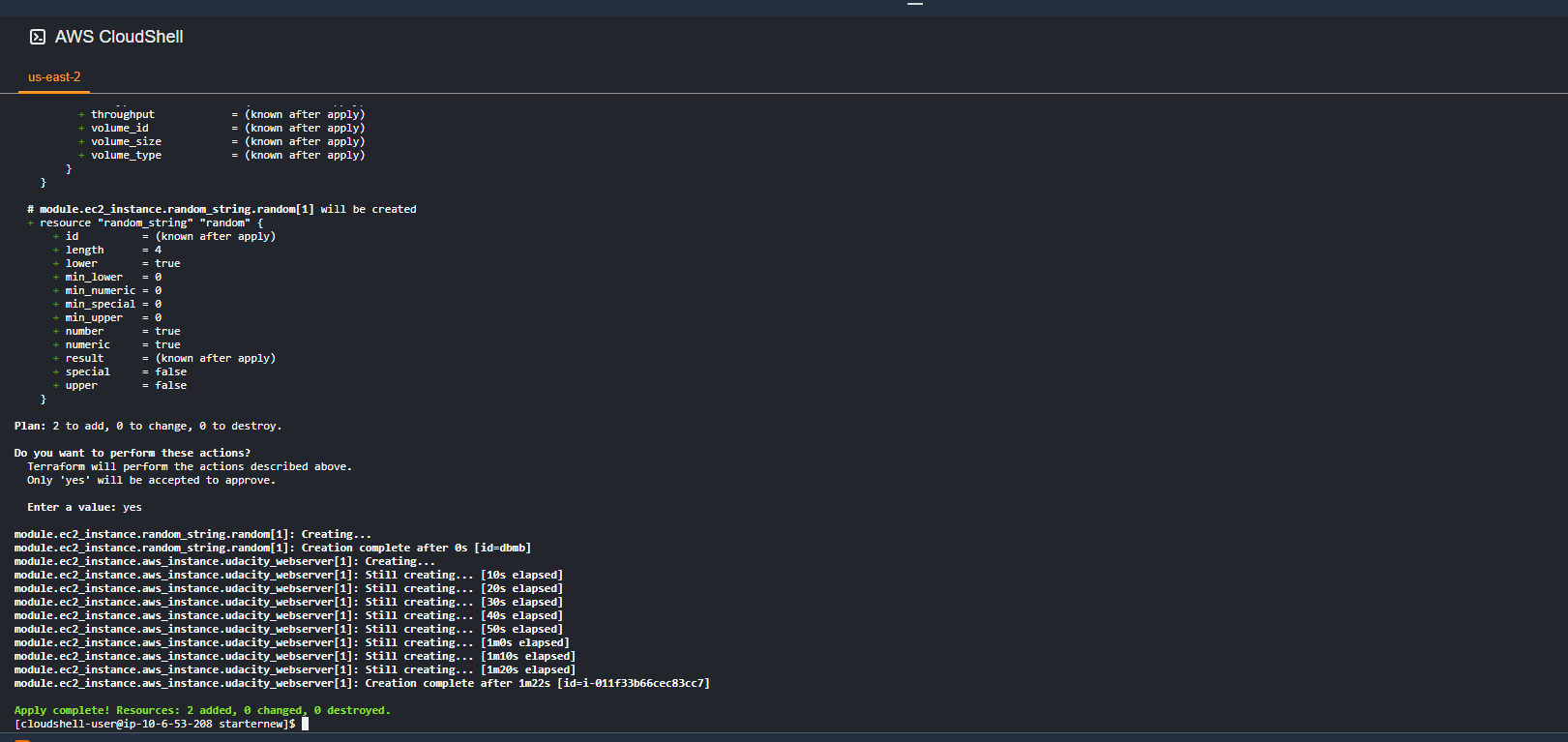
Adding 3 Virtual machines to us-east-2 region(Zone1)



**A**

Adding ALB to Zone1 us-east-2

# module.alb.aws\_lb\_target\_group.my-target-group will be created

+ resource "aws\_lb\_target\_group" "my-target-group" {

+ arn = (known after apply)

+ arn\_suffix = (known after apply)

+ connection\_termination = false

+ deregistration\_delay = "300"

+ id = (known after apply)

+ lambda\_multi\_value\_headers\_enabled = false

+ load\_balancing\_algorithm\_type = (known after apply)

+ name = "my-test-tg"

+ port = 80

+ preserve\_client\_ip = (known after apply)

+ protocol = "HTTP"

+ protocol\_version = (known after apply)

+ proxy\_protocol\_v2 = false

+ slow\_start = 0

+ tags\_all = (known after apply)

+ target\_type = "instance"

+ vpc\_id = "vpc-077baef119aed7908"

+ health\_check {

+ enabled = true

+ healthy\_threshold = 5

+ interval = 10

+ matcher = (known after apply)

+ path = "/"

+ port = "traffic-port"

+ protocol = "HTTP"

+ timeout = 5

+ unhealthy\_threshold = 2

}

+ stickiness {

+ cookie\_duration = (known after apply)

+ cookie\_name = (known after apply)

+ enabled = (known after apply)

+ type = (known after apply)

}

}

# module.alb.aws\_lb\_target\_group\_attachment.my-alb-target-group-attachment1 will be created

+ resource "aws\_lb\_target\_group\_attachment" "my-alb-target-group-attachment1" {

+ id = (known after apply)

+ port = 80

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

+ target\_id = "appserver-instance-aw5y9"

}

# module.alb.aws\_lb\_target\_group\_attachment.my-alb-target-group-attachment2 will be created

+ resource "aws\_lb\_target\_group\_attachment" "my-alb-target-group-attachment2" {

+ id = (known after apply)

+ port = 80

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

+ target\_id = "appserver-instance-awdbmb"

}

# module.alb.aws\_security\_group.my-alb-sg will be created

+ resource "aws\_security\_group" "my-alb-sg" {

+ arn = (known after apply)

+ description = "Managed by Terraform"

+ egress = (known after apply)

+ id = (known after apply)

+ ingress = (known after apply)

+ name = "my-alb-sg"

+ name\_prefix = (known after apply)

+ owner\_id = (known after apply)

+ revoke\_rules\_on\_delete = false

+ tags\_all = (known after apply)

+ vpc\_id = "vpc-077baef119aed7908"

}

# module.alb.aws\_security\_group\_rule.inbound\_http will be created

+ resource "aws\_security\_group\_rule" "inbound\_http" {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ from\_port = 80

+ id = (known after apply)

+ protocol = "tcp"

+ security\_group\_id = (known after apply)

+ self = false

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

+ to\_port = 80

+ type = "ingress"

}

# module.alb.aws\_security\_group\_rule.inbound\_ssh will be created

+ resource "aws\_security\_group\_rule" "inbound\_ssh" {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ from\_port = 22

+ id = (known after apply)

+ protocol = "tcp"

+ security\_group\_id = (known after apply)

+ self = false

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

+ to\_port = 22

+ type = "ingress"

}

# module.alb.aws\_security\_group\_rule.outbound\_all will be created

+ resource "aws\_security\_group\_rule" "outbound\_all" {

+ cidr\_blocks = [

+ "0.0.0.0/0",

]

+ from\_port = 0

+ id = (known after apply)

+ protocol = "-1"

+ security\_group\_id = (known after apply)

+ self = false

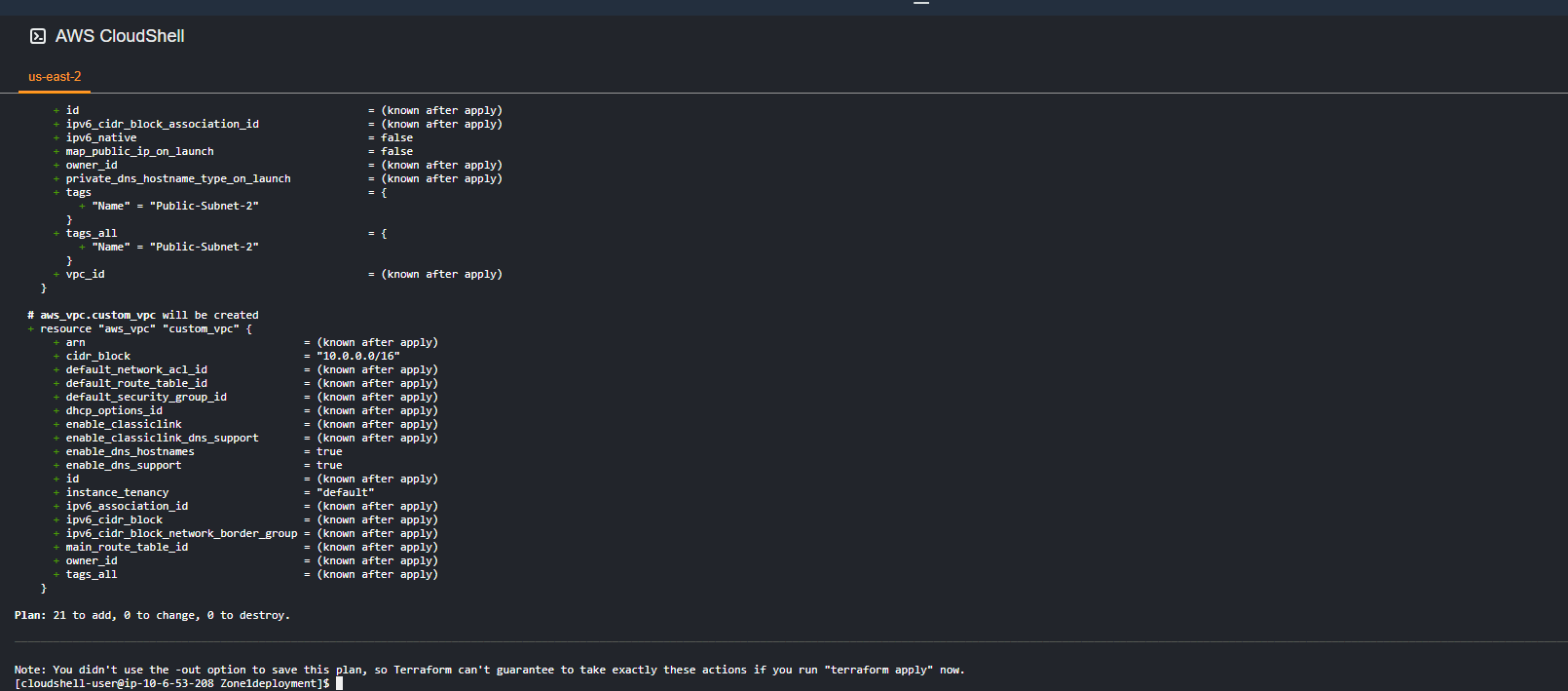
+ source\_security\_group\_id = (known after apply)

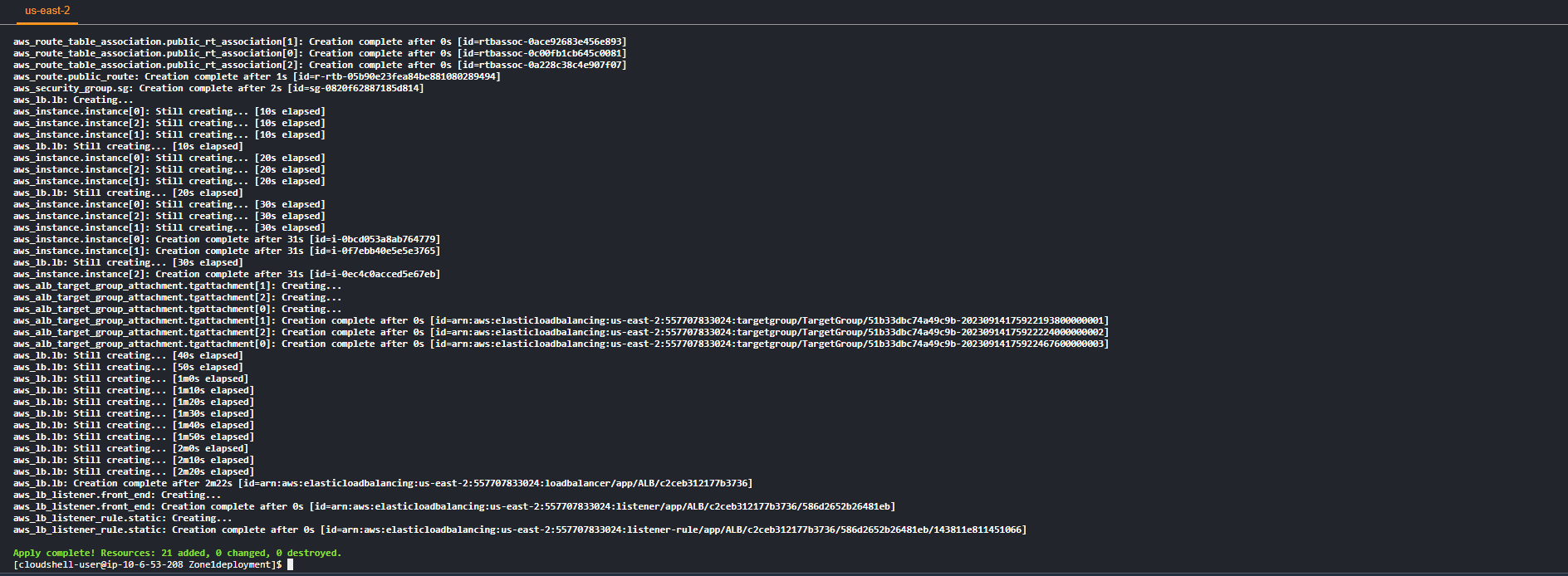
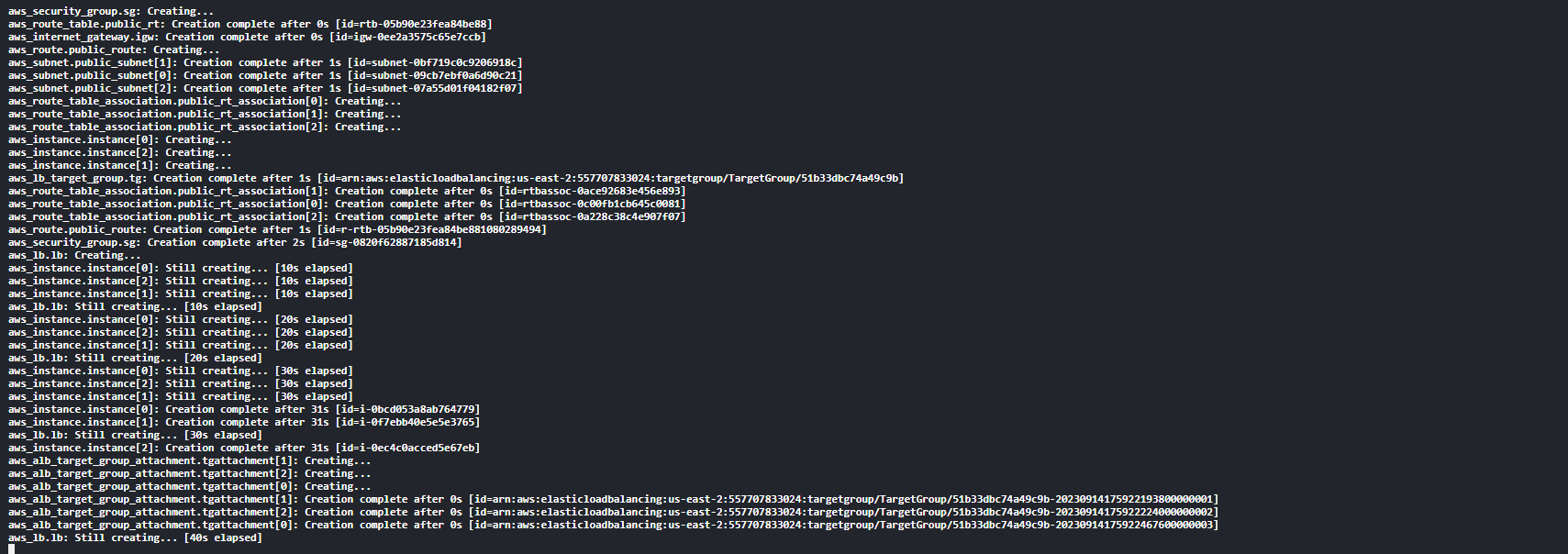
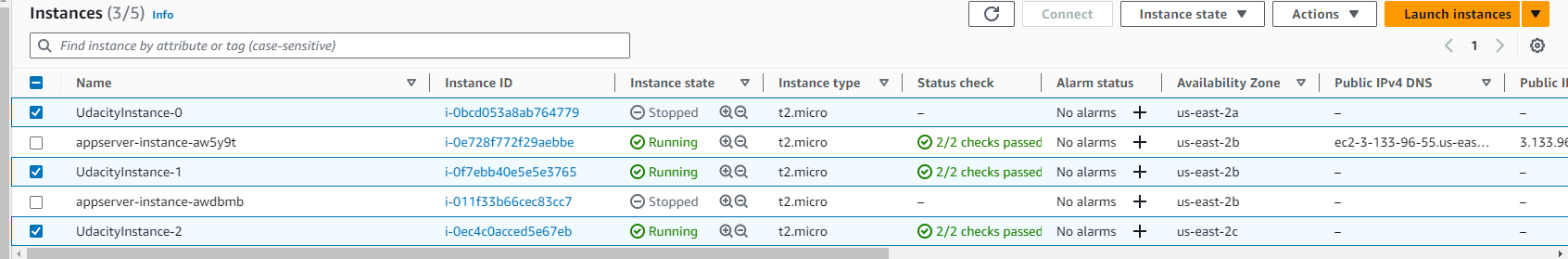
+ to\_port = 0

+ type = "egress"

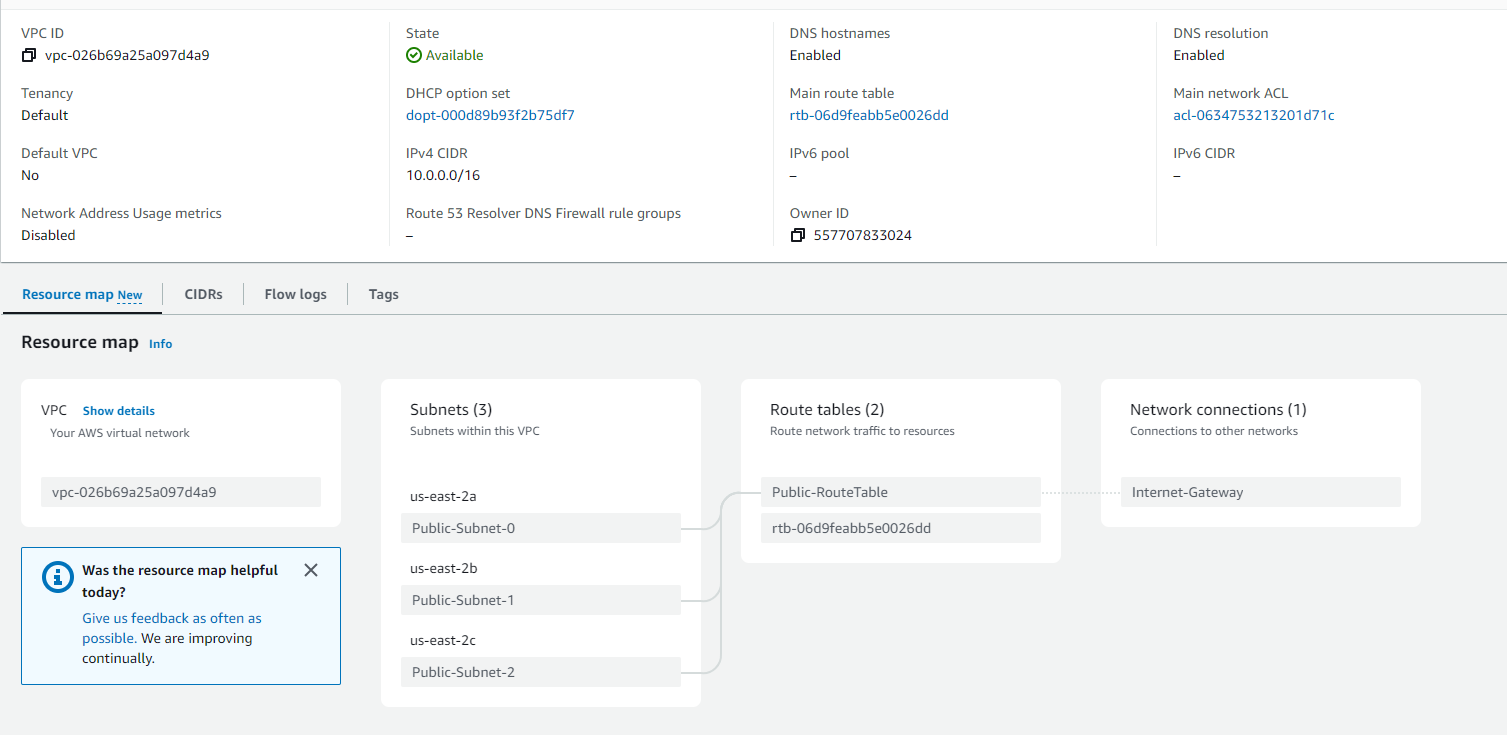
}

Terraform code status



VPC has ip in multiple Az



ALB setup

