Problem Statement - Create an Autoscaling group using Terraform on AWS. The instances in the group should have Nginx installed on them. Parameters in the ASG can be set at your discretion and should be mentioned in the solution documentation.
Any variables in the script should be placed in a separate variables.tf file. The output of the script should be the DNS of the associated load balancer.

Grading Criteria

1)	Documentation explaining the solution architecture and template created -	10 marks
2)	Shell script used to install Nginx	5 marks
3)	Separate variables file	5 marks
4)	Output file showing DNS of load balancer	10 marks
5)	Terraform script with given components	
	a) Security groups	2 marks
	b) EC2 instances	3 marks
	c) Elastic Load balancer	5 marks
	d) Autoscaling group	5 marks
6)	Screenshot showing successful execution of Terraform script	5 marks

---

Solution

1) The architecture contains a VPC with two public subnets in different regions (us-east-1a and us-east-1b), one private subnet (us-east-1c), one internet gateway, a custom route table with destination as “anywhere” and target as internet gateway associated with two public subnets. It also contains a NAT gateway in public subnet 1 associated with private subnet through a custom route table for private subnet separately.

   The instances in the public subnets are maintained by autoscaling group. It launches/terminates the instances depending on the auto scaling group setting and autoscaling group policies. The instances have a predefined launch configuration as described in aws_launch_configuration.

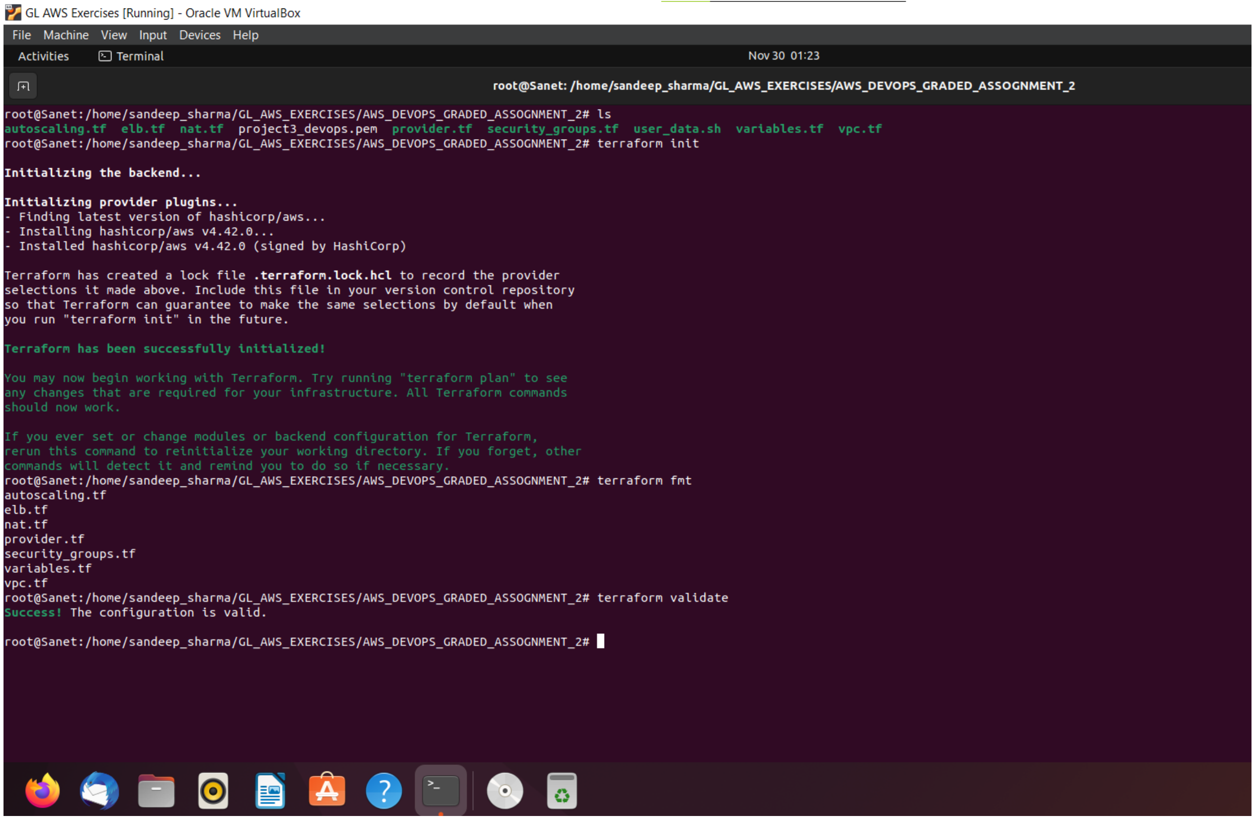
   There is an elastic load balancer which distributes the traffic between the instances launched by autoscaling group.

   There are two security groups defined, for instances in public subnets (inbound : port 80 – http, port 22 – ssh || outbound : anywhere) and for elastic load balancer (inbound : port 80 – http || outbound : anywhere). Both security groups have been associated with VPC.

FILES:

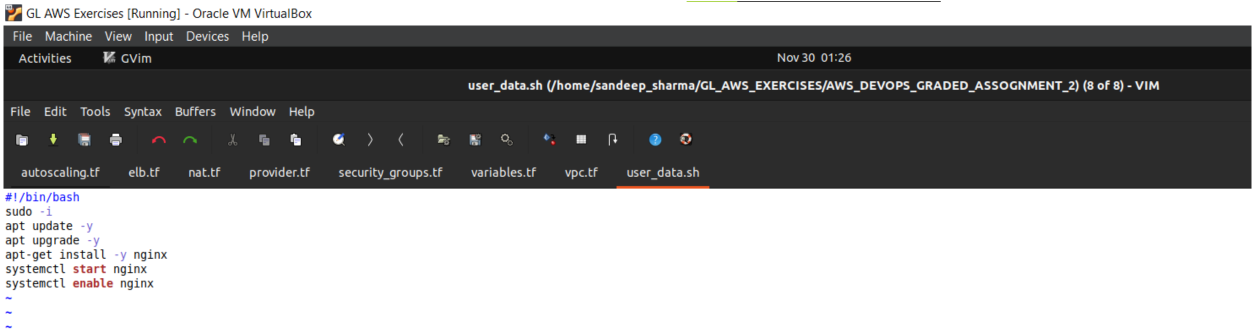
a) autoscaling.tf – contains “aws_launch_configuration”, “aws_autoscaling_group”, “aws_autoscaling_policy” for scaleup and scaledown.
b) elb.tf – has “aws_elb” (Elastic Load Balancer) and “output” defined.
c) nat.tf – cotains “aws_nat_gateway”, “aws_route_table” for private subnet and nat gateway connection, “aws_route_table_association” to associate route table to private subnet and “aws_eip” to give an IP to nat gateway after creation.
d) provider.tf – defines which cloud provider plugin to use while terraform executes.
e) Security_groups.tf – has security groups “aws_security_group” defined for instances and elb.
f) user_data.sh – a shell script to update the libraries and setup nginx after instances have been created and launched.
g) vpc.tf – contains “aws_vpc”, “aws_subnet” for 2 public and 1 private, “aws_internet_gateway”, “aws_route_table” for 2 public subnets, and “aws_route_table_association” for both public subnets to get associated with the route_table.
h) variables.tf – defines variables used in other resources and assigns their values.

Initial setup:


GL AWS Exercises [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Nov 30 01:23
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

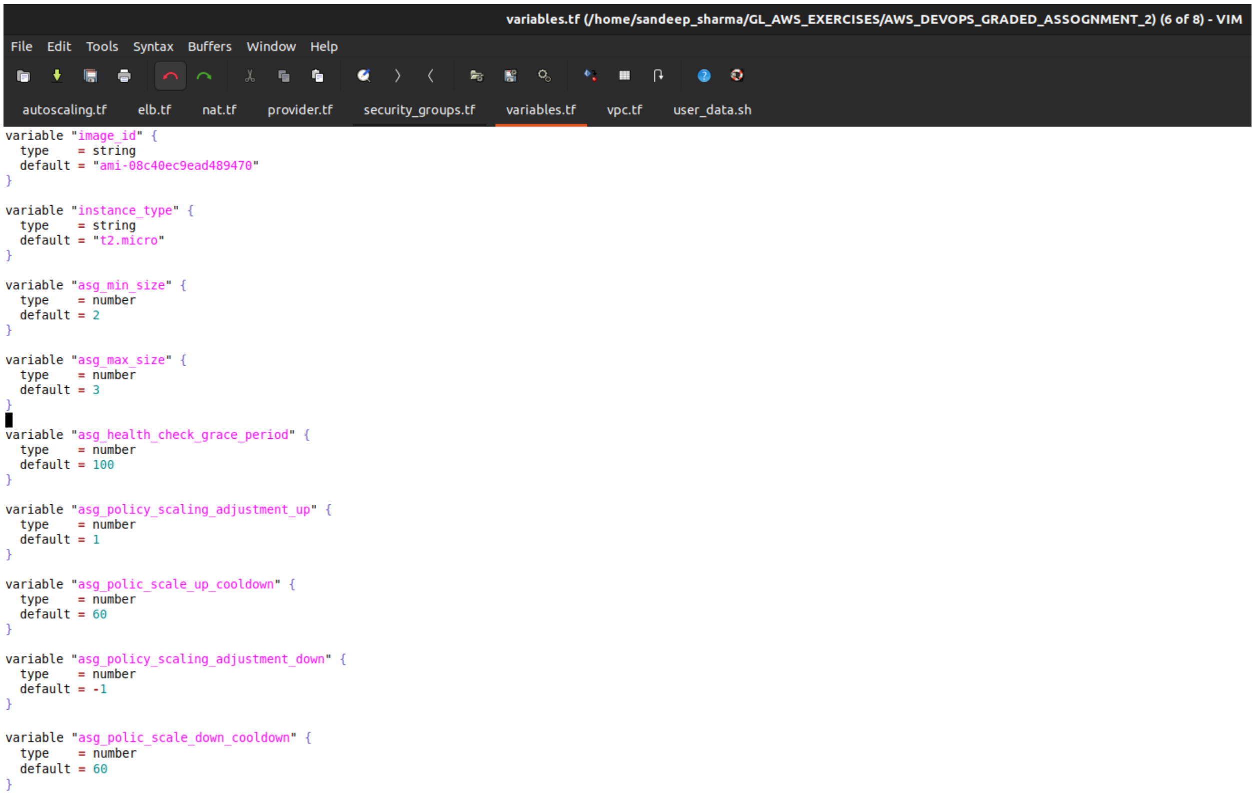
```
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# ls
autoscaling.tf  elb.tf  nat.tf  project3_devops.pem  provider.tf  security_groups.tf  user_data.sh  variables.tf  vpc.tf
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v4.42.0...
- Installed hashicorp/aws v4.42.0 (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@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform fmt
autoscaling.tf
elb.tf
nat.tf
provider.tf
security_groups.tf
variables.tf
vpc.tf
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform validate
Success! The configuration is valid.

root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# 
```

2)


GL AWS Exercises [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
GVim
Nov 30 01:26
user_data.sh (/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2) (8 of 8) - VIM
File Edit Tools Syntax Buffers Window Help
autoscaling.tf
elb.tf
nat.tf
provider.tf
security_groups.tf
variables.tf
vpc.tf
user_data.sh

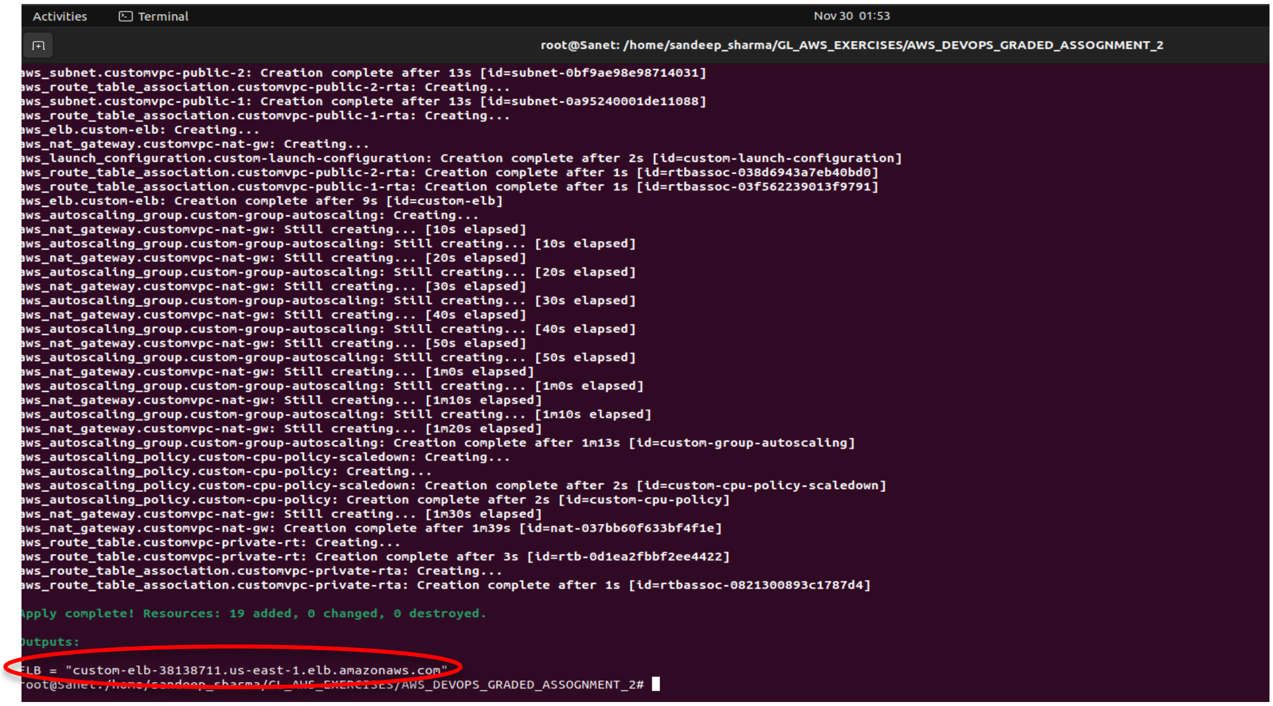
```
#!/bin/bash
sudo -i
apt update -y
apt upgrade -y
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
~
~
~
```

3)


variables.tf (/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2) (6 of 8) - VIM
File Edit Tools Syntax Buffers Window Help
autoscaling.tf
elb.tf
nat.tf
provider.tf
security_groups.tf
variables.tf
vpc.tf
user_data.sh

```
variable "image_id" {
  type    = string
  default = "ami-08c40ec9ead489470"
}

variable "instance_type" {
  type    = string
  default = "t2.micro"
}

variable "asg_min_size" {
  type    = number
  default = 2
}

variable "asg_max_size" {
  type    = number
  default = 3
}

variable "asg_health_check_grace_period" {
  type    = number
  default = 100
}

variable "asg_policy_scaling_adjustment_up" {
  type    = number
  default = 1
}

variable "asg_polic_scale_up_cooldown" {
  type    = number
  default = 60
}

variable "asg_policy_scaling_adjustment_down" {
  type    = number
  default = -1
}

variable "asg_polic_scale_down_cooldown" {
  type    = number
  default = 60
}
```

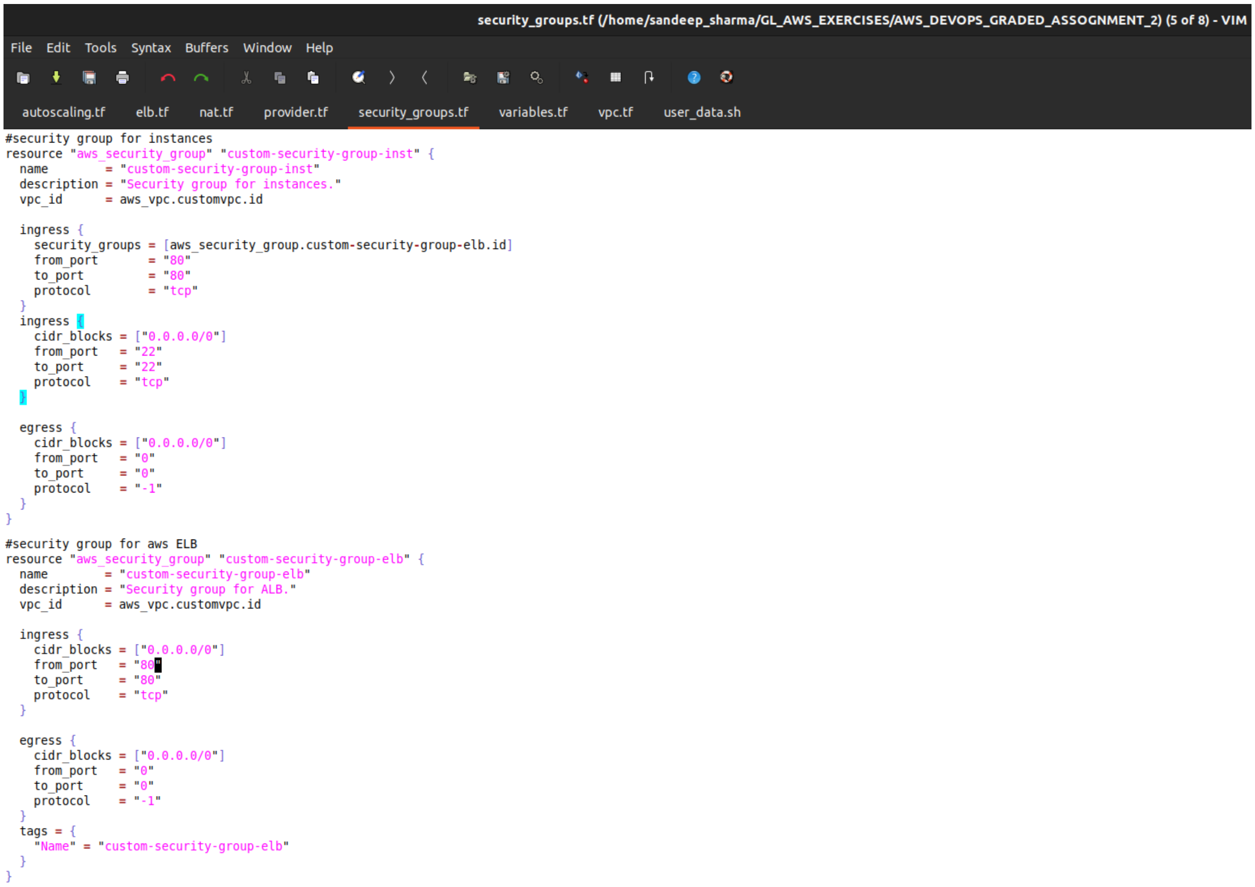
4)


Activities
Terminal
Nov 30 01:53
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
aws_subnet.customvpc-public-2: Creation complete after 13s [id=subnet-0bf9ae98e98714031]
aws_route_table_association.customvpc-public-2-rta: Creating...
aws_subnet.customvpc-public-1: Creation complete after 13s [id=subnet-0a95240001de11088]
aws_route_table_association.customvpc-public-1-rta: Creating...
aws_elb.custom-elb: Creating...
aws_nat_gateway.customvpc-nat-gw: Creating...
aws_launch_configuration.custom-launch-configuration: Creation complete after 2s [id=custom-launch-configuration]
aws_route_table_association.customvpc-public-2-rta: Creation complete after 1s [id=rtbassoc-038d6943a7eb40bd0]
aws_route_table_association.customvpc-public-1-rta: Creation complete after 1s [id=rtbassoc-03f562239013f9791]
aws_elb.custom-elb: Creation complete after 9s [id=custom-elb]
aws_autoscaling_group.custom-group-autoscaling: Creating...
aws_nat_gateway.customvpc-nat-gw: Still creating... [10s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [10s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [20s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [30s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [30s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [40s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [40s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [50s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [50s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m0s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [1m0s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m10s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [1m10s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Creation complete after 1m13s [id=custom-group-autoscaling]
aws_autoscaling_policy.custom-cpu-policy-scaledown: Creating...
aws_autoscaling_policy.custom-cpu-policy: Creating...
aws_autoscaling_policy.custom-cpu-policy-scaledown: Creation complete after 2s [id=custom-cpu-policy-scaledown]
aws_autoscaling_policy.custom-cpu-policy: Creation complete after 2s [id=custom-cpu-policy]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m30s elapsed]
aws_nat_gateway.customvpc-nat-gw: Creation complete after 1m39s [id=nat-037bb60f633bf4f1e]
aws_route_table.customvpc-private-rt: Creating...
aws_route_table.customvpc-private-rt: Creation complete after 3s [id=rtb-0d1ea2fbbf2ee4422]
aws_route_table_association.customvpc-private-rta: Creating...
aws_route_table_association.customvpc-private-rta: Creation complete after 1s [id=rtbassoc-0821300893c1787d4]

Apply complete! Resources: 19 added, 0 changed, 0 destroyed.

Outputs:

ELB = "custom-elb-38138711.us-east-1.elb.amazonaws.com"
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2#
```

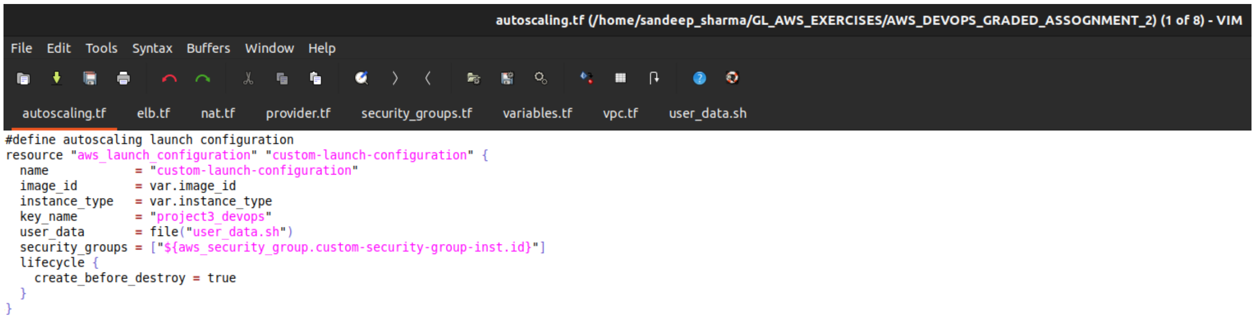
5) Terraform scripts:

a. Security groups:

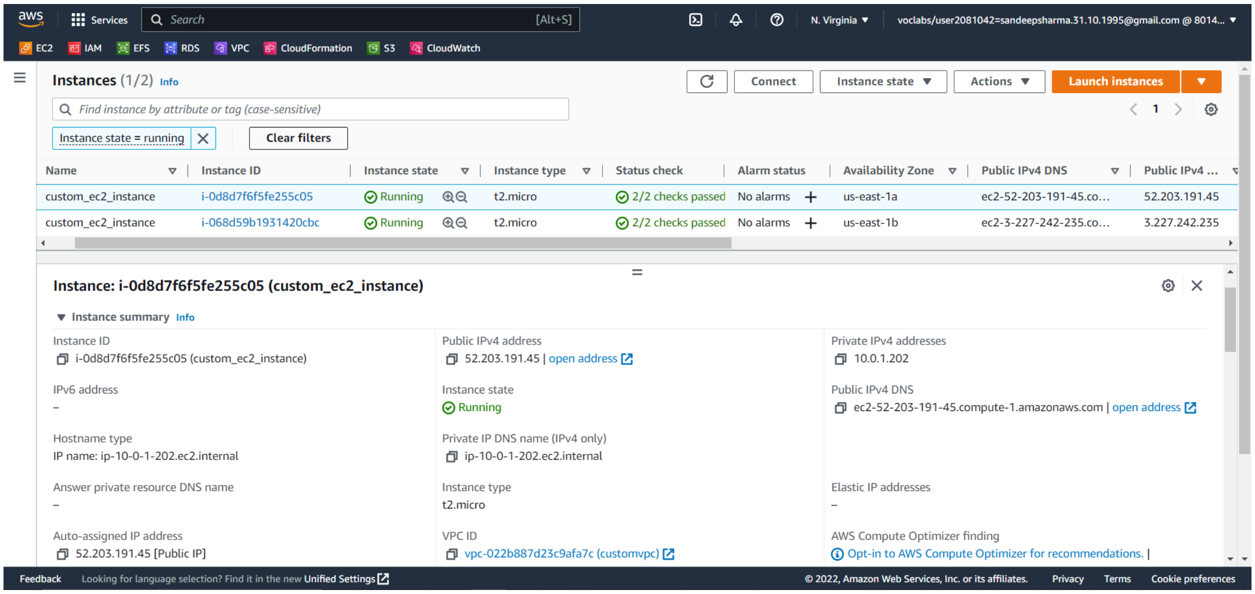

security_groups.tf (/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2) (5 of 8) - VIM
File Edit Tools Syntax Buffers Window Help
autoscaling.tf elb.tf nat.tf provider.tf security_groups.tf variables.tf vpc.tf user_data.sh

```
#security group for instances
resource "aws_security_group" "custom-security-group-inst" {
  name        = "custom-security-group-inst"
  description = "Security group for instances."
  vpc_id      = aws_vpc.customvpc.id

  ingress {
    security_groups = [aws_security_group.custom-security-group-elb.id]
    from_port       = "80"
    to_port         = "80"
    protocol        = "tcp"
  }
  ingress {
    cidr_blocks = ["0.0.0.0/0"]
    from_port   = "22"
    to_port     = "22"
    protocol    = "tcp"
  }

  egress {
    cidr_blocks = ["0.0.0.0/0"]
    from_port   = "0"
    to_port     = "0"
    protocol    = "-1"
  }
}

#security group for aws ELB
resource "aws_security_group" "custom-security-group-elb" {
  name        = "custom-security-group-elb"
  description = "Security group for ALB."
  vpc_id      = aws_vpc.customvpc.id

  ingress {
    cidr_blocks = ["0.0.0.0/0"]
    from_port   = "80"
    to_port     = "80"
    protocol    = "tcp"
  }

  egress {
    cidr_blocks = ["0.0.0.0/0"]
    from_port   = "0"
    to_port     = "0"
    protocol    = "-1"
  }
  tags = {
    "Name" = "custom-security-group-elb"
  }
}
```

b. EC2 instances (instances are launched by auto scaling group itself.. I have not created aws_instancce separately) – created using launch configuration


autoscaling.tf (/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2) (1 of 8) - VIM
File Edit Tools Syntax Buffers Window Help
autoscaling.tf elb.tf nat.tf provider.tf security_groups.tf variables.tf vpc.tf user_data.sh

```
#define autoscaling launch configuration
resource "aws_launch_configuration" "custom-launch-configuration" {
  name            = "custom-launch-configuration"
  image_id        = var.image_id
  instance_type   = var.instance_type
  key_name        = "project3_devops"
  user_data       = file("user_data.sh")
  security_groups = ["${aws_security_group.custom-security-group-inst.id}"]
  lifecycle {
    create_before_destroy = true
  }
}
```


aws
Services
Search
[Alt+S]
N. Virginia
voclabs/user2081042=sandeepsharma.31.10.1995@gmail.com @ 8014...
EC2
IAM
EFS
RDS
VPC
CloudFormation
S3
CloudWatch
Connect
Instance state
Actions
Launch instances
Find instance by attribute or tag (case-sensitive)
Clear filters
1
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Instances (1/2) Info

Instance state = running

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
custom_ec2_instance	i-0d8d7f6f5fe255c05	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-east-1a	ec2-52-203-191-45.co...	52.203.191.45
custom_ec2_instance	i-068d59b1931420cbc	✓ Running	t2.micro	✓ 2/2 checks passed	No alarms +	us-east-1b	ec2-3-227-242-235.co...	3.227.242.235

Instance: i-0d8d7f6f5fe255c05 (custom_ec2_instance)

▼ Instance summary Info

Instance ID
i-0d8d7f6f5fe255c05 (custom_ec2_instance)

Public IPv4 address
52.203.191.45 | open address

Private IPv4 addresses
10.0.1.202

IPv6 address
–

Instance state
✓ Running

Public IPv4 DNS
ec2-52-203-191-45.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-10-0-1-202.ec2.internal

Private IP DNS name (IPv4 only)
ip-10-0-1-202.ec2.internal

Answer private resource DNS name
–

Instance type
t2.micro

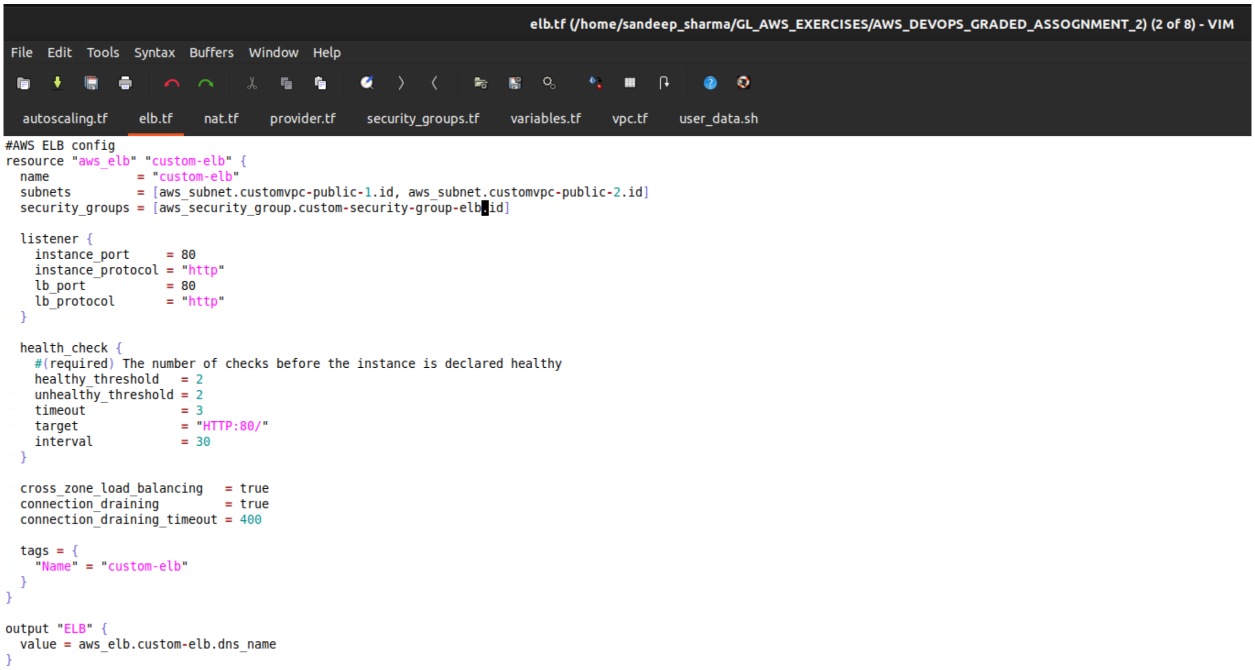
Elastic IP addresses
–

Auto-assigned IP address
52.203.191.45 [Public IP]

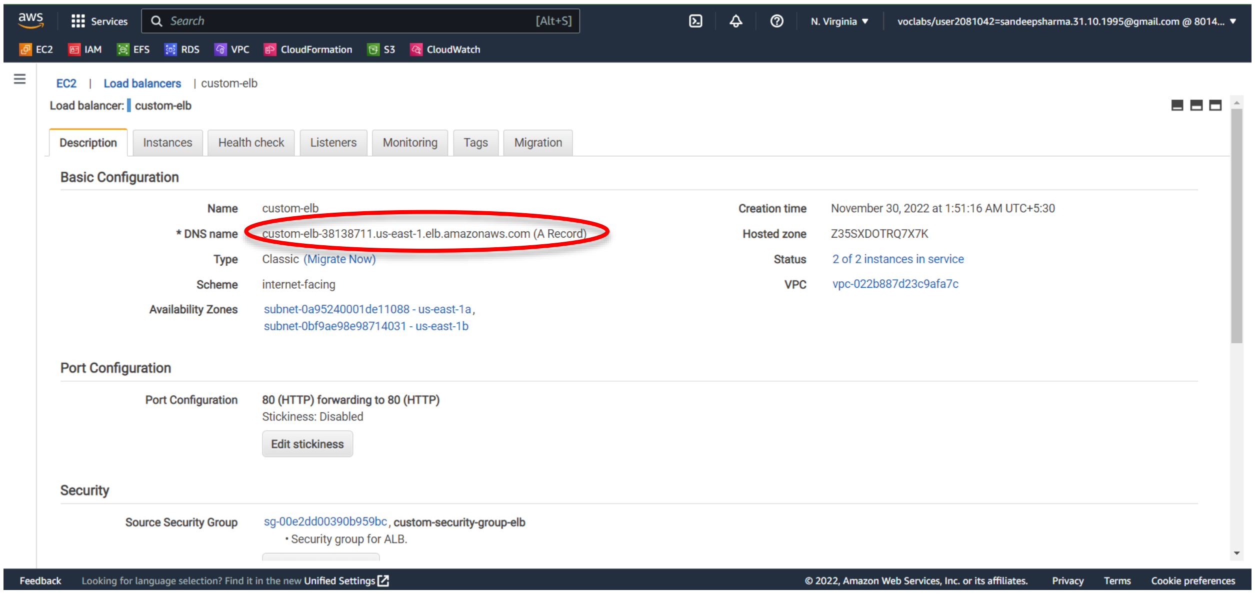
VPC ID
vpc-022b887d23c9afa7c (customvpc)

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. |

c.



```
#AWS ELB config
resource "aws_elb" "custom-elb" {
  name            = "custom-elb"
  subnets         = [aws_subnet.customvpc-public-1.id, aws_subnet.customvpc-public-2.id]
  security_groups = [aws_security_group.custom-security-group-elb.id]

  listener {
    instance_port     = 80
    instance_protocol = "http"
    lb_port           = 80
    lb_protocol       = "http"
  }

  health_check {
    #(required) The number of checks before the instance is declared healthy
    healthy_threshold   = 2
    unhealthy_threshold = 2
    timeout             = 3
    target              = "HTTP:80/"
    interval            = 30
  }

  cross_zone_load_balancing   = true
  connection_draining         = true
  connection_draining_timeout = 400

  tags = {
    "Name" = "custom-elb"
  }
}

output "ELB" {
  value = aws_elb.custom-elb.dns_name
}
```


aws
Services
Search
[Alt+S]
N. Virginia ▼
voclabs/user2081042=sandeepsharma.31.10.1995@gmail.com @ 8014... ▼
EC2
IAM
EFS
RDS
VPC
CloudFormation
S3
CloudWatch
EC2 | Load balancers | custom-elb
Description
Instances
Health check
Listeners
Monitoring
Tags
Migration
Edit stickiness
Feedback
Looking for language selection? Find it in the new Unified Settings
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Load balancer: custom-elb

Basic Configuration

Name: custom-elb
* DNS name: custom-elb-38138711.us-east-1.elb.amazonaws.com (A Record)
Type: Classic (Migrate Now)
Scheme: internet-facing
Availability Zones: subnet-0a95240001de11088 - us-east-1a, subnet-0bf9ae98e98714031 - us-east-1b

Creation time: November 30, 2022 at 1:51:16 AM UTC+5:30
Hosted zone: Z35SXDOTRQ7X7K
Status: 2 of 2 instances in service
VPC: vpc-022b887d23c9afa7c

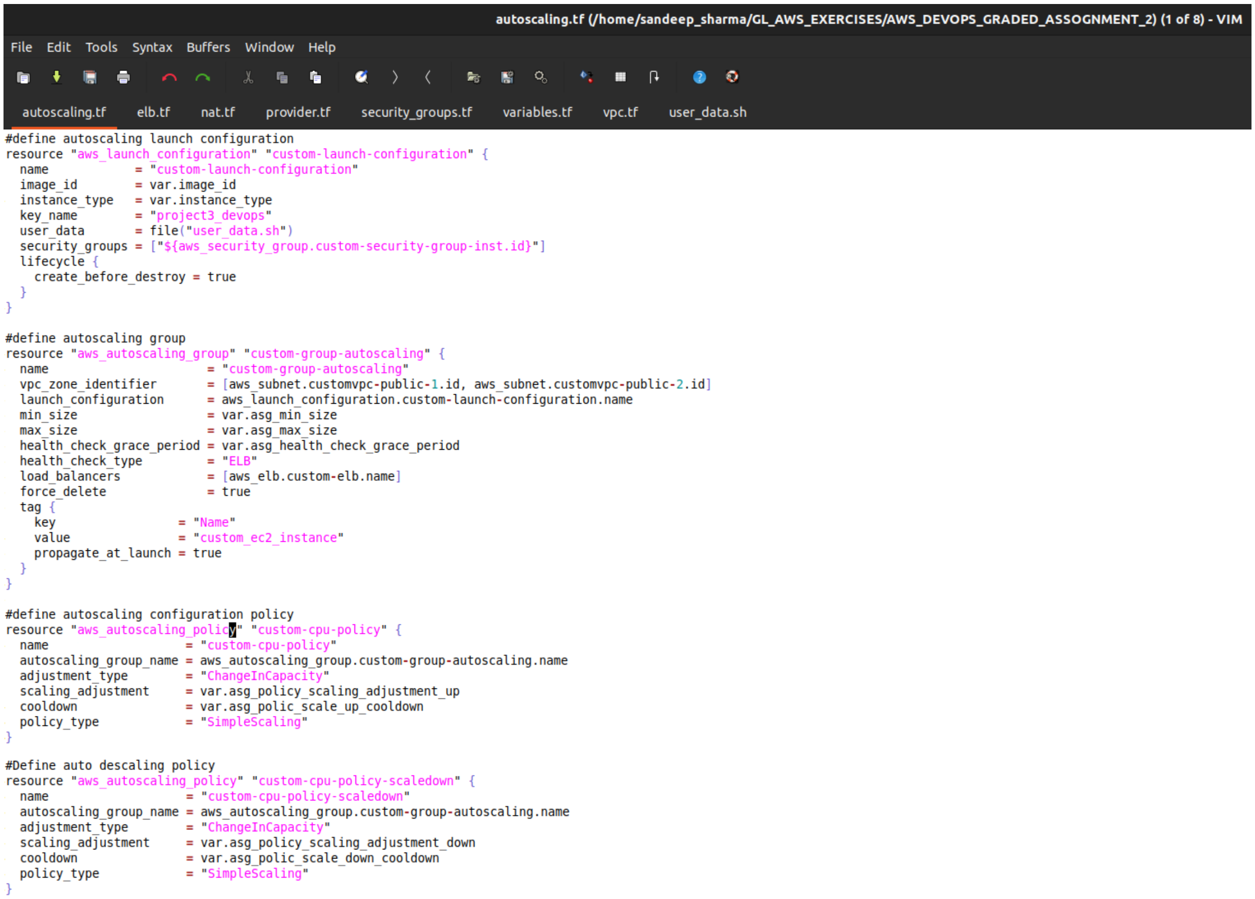
Port Configuration

Port Configuration: 80 (HTTP) forwarding to 80 (HTTP)
Stickiness: Disabled

Security

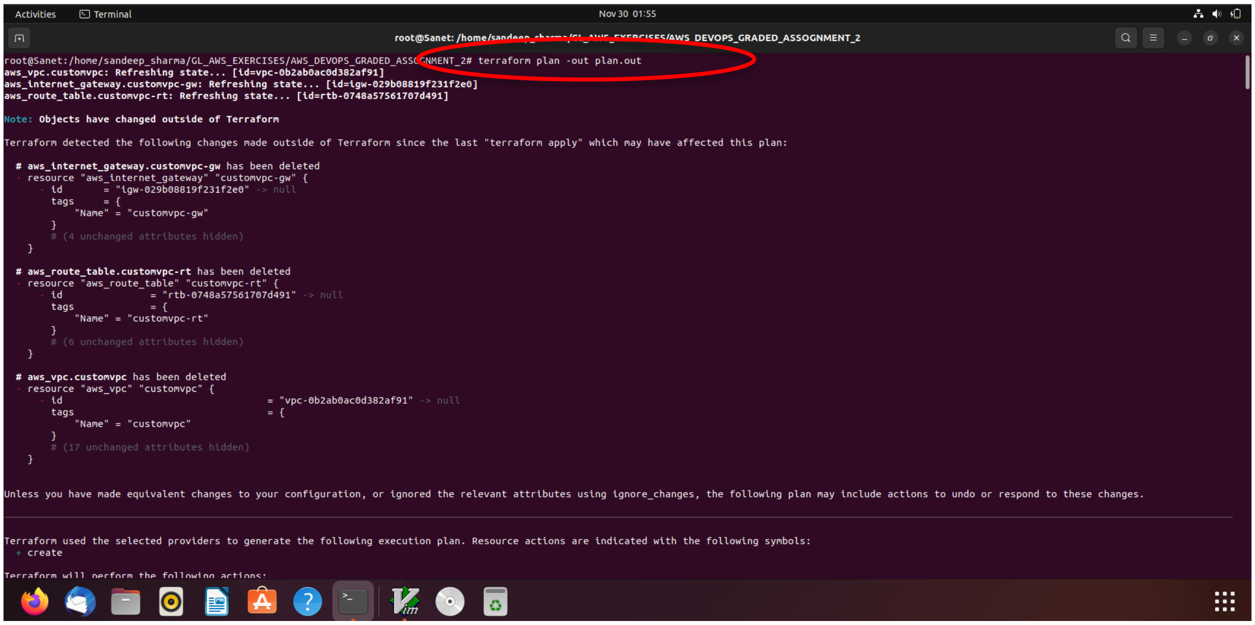
Source Security Group: sg-00e2dd00390b959bc, custom-security-group-elb
- Security group for ALB.

d.

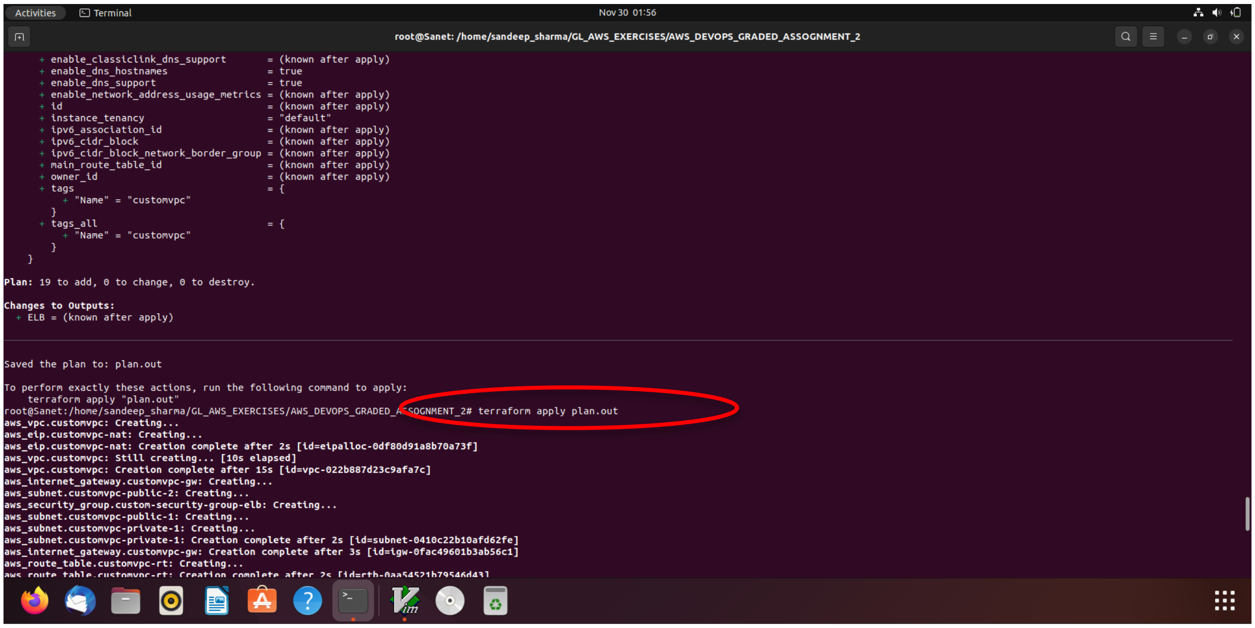


```
#define autoscaling launch configuration
resource "aws_launch_configuration" "custom-launch-configuration" {
  name            = "custom-launch-configuration"
  image_id        = var.image_id
  instance_type   = var.instance_type
  key_name        = "project3_devops"
  user_data       = file("user_data.sh")
  security_groups = ["${aws_security_group.custom-security-group-inst.id}"]
  lifecycle {
    create_before_destroy = true
  }
}

#define autoscaling group
resource "aws_autoscaling_group" "custom-group-autoscaling" {
  name                      = "custom-group-autoscaling"
  vpc_zone_identifier       = [aws_subnet.customvpc-public-1.id, aws_subnet.customvpc-public-2.id]
  launch_configuration      = aws_launch_configuration.custom-launch-configuration.name
  min_size                  = var.asg_min_size
  max_size                  = var.asg_max_size
  health_check_grace_period = var.asg_health_check_grace_period
  health_check_type         = "ELB"
  load_balancers            = [aws_elb.custom-elb.name]
  force_delete              = true
  tag {
    key                 = "Name"
    value               = "custom_ec2_instance"
    propagate_at_launch = true
  }
}

#define autoscaling configuration policy
resource "aws_autoscaling_policy" "custom-cpu-policy" {
  name                   = "custom-cpu-policy"
  autoscaling_group_name = aws_autoscaling_group.custom-group-autoscaling.name
  adjustment_type        = "ChangeInCapacity"
  scaling_adjustment     = var.asg_policy_scaling_adjustment_up
  cooldown               = var.asg_polic_scale_up_cooldown
  policy_type            = "SimpleScaling"
}

#Define auto descaling policy
resource "aws_autoscaling_policy" "custom-cpu-policy-scaledown" {
  name                   = "custom-cpu-policy-scaledown"
  autoscaling_group_name = aws_autoscaling_group.custom-group-autoscaling.name
  adjustment_type        = "ChangeInCapacity"
  scaling_adjustment     = var.asg_policy_scaling_adjustment_down
  cooldown               = var.asg_polic_scale_down_cooldown
  policy_type            = "SimpleScaling"
}
```

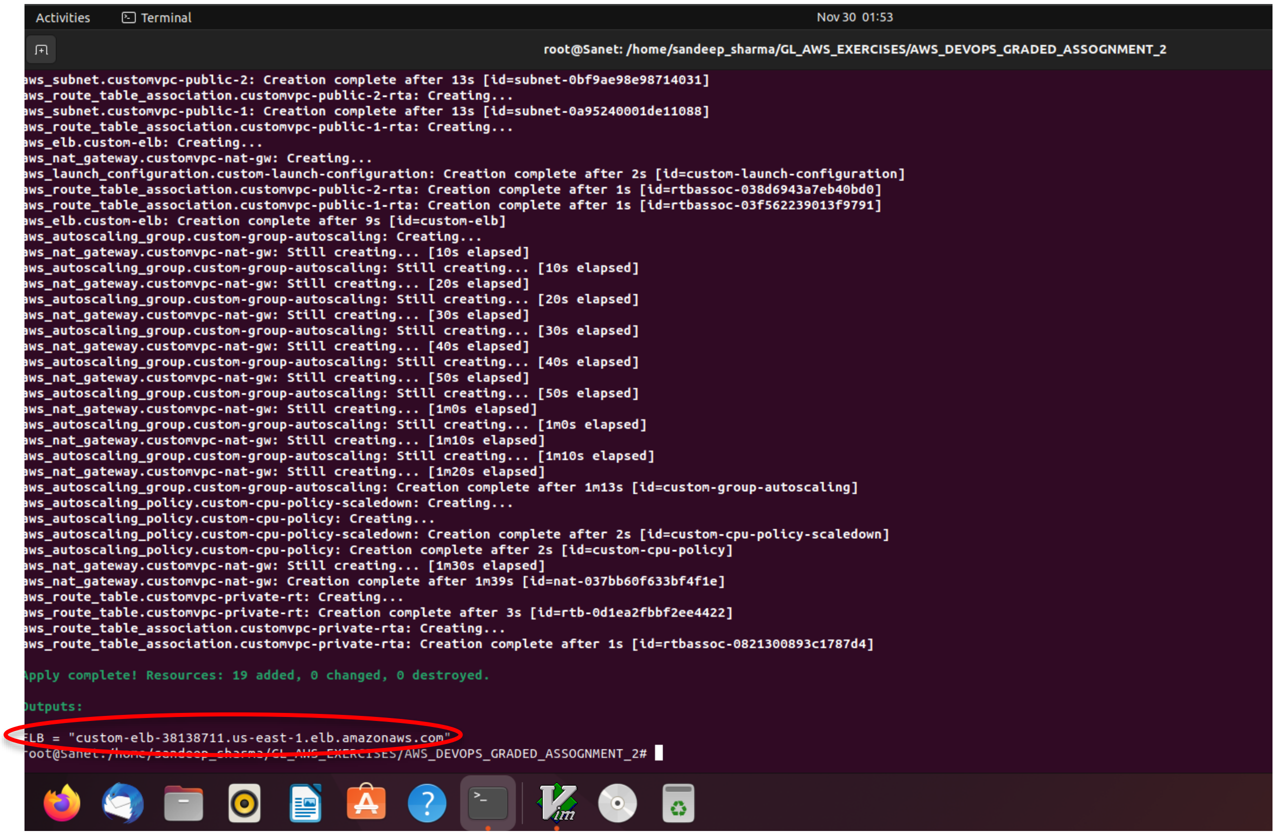
6)


Activities
Terminal
Nov 30 01:55
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

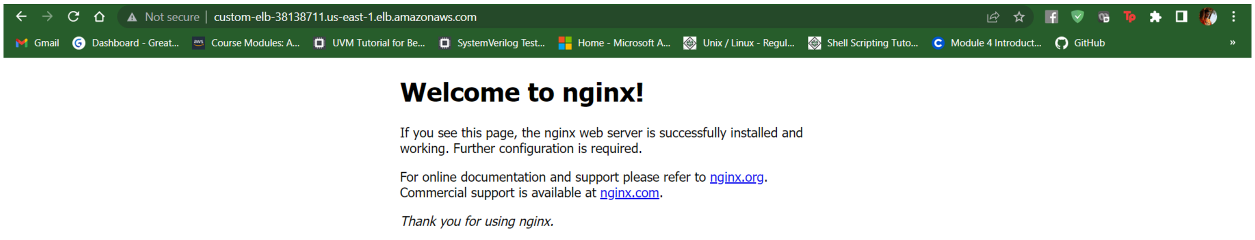
```
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform plan -out plan.out
aws_vpc.customvpc: Refreshing state... [id=vpc-0b2ab0ac0d382af91]
aws_internet_gateway.customvpc-gw: Refreshing state... [id=igw-029b08819f231f2e0]
aws_route_table.customvpc-rt: Refreshing state... [id=rtb-0748a57561707d491]

Note: Objects have changed outside of Terraform

Terraform detected the following changes made outside of Terraform since the last "terraform apply" which may have affected this plan:

  # aws_internet_gateway.customvpc-gw has been deleted
  - resource "aws_internet_gateway" "customvpc-gw" {
      - id       = "igw-029b08819f231f2e0" -> null
        tags     = {
            "Name" = "customvpc-gw"
        }
        # (4 unchanged attributes hidden)
    }

  # aws_route_table.customvpc-rt has been deleted
  - resource "aws_route_table" "customvpc-rt" {
      - id               = "rtb-0748a57561707d491" -> null
        tags             = {
            "Name" = "customvpc-rt"
        }
        # (6 unchanged attributes hidden)
    }

  # aws_vpc.customvpc has been deleted
  - resource "aws_vpc" "customvpc" {
      - id                                   = "vpc-0b2ab0ac0d382af91" -> null
        tags                                 = {
            "Name" = "customvpc"
        }
        # (17 unchanged attributes hidden)
    }


Unless you have made equivalent changes to your configuration, or ignored the relevant attributes using ignore_changes, the following plan may include actions to undo or respond to these changes.

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

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:
```


Activities
Terminal
Nov 30 01:56
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
      + enable_classiclink_dns_support       = (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" = "customvpc"
        }
      + tags_all                             = {
          + "Name" = "customvpc"
        }
    }

Plan: 19 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + ELB = (known after apply)

Saved the plan to: plan.out

To perform exactly these actions, run the following command to apply:
    terraform apply "plan.out"
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform apply plan.out
aws_vpc.customvpc: Creating...
aws_eip.customvpc-nat: Creating...
aws_eip.customvpc-nat: Creation complete after 2s [id=eipalloc-0df80d91a8b70a73f]
aws_vpc.customvpc: Still creating... [10s elapsed]
aws_vpc.customvpc: Creation complete after 15s [id=vpc-022b887d23c9afa7c]
aws_internet_gateway.customvpc-gw: Creating...
aws_subnet.customvpc-public-2: Creating...
aws_security_group.custom-security-group-elb: Creating...
aws_subnet.customvpc-public-1: Creating...
aws_subnet.customvpc-private-1: Creating...
aws_subnet.customvpc-private-1: Creation complete after 2s [id=subnet-0410c22b10afd62fe]
aws_internet_gateway.customvpc-gw: Creation complete after 3s [id=igw-0fac49601b3ab56c1]
aws_route_table.customvpc-rt: Creating...
aws_route_table.customvpc-rt: Creation complete after 2s [id=rtb-0aa54521b79546d43]
```


Activities
Terminal
Nov 30 01:53
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
aws_subnet.customvpc-public-2: Creation complete after 13s [id=subnet-0bf9ae98e98714031]
aws_route_table_association.customvpc-public-2-rta: Creating...
aws_subnet.customvpc-public-1: Creation complete after 13s [id=subnet-0a95240001de11088]
aws_route_table_association.customvpc-public-1-rta: Creating...
aws_elb.custom-elb: Creating...
aws_nat_gateway.customvpc-nat-gw: Creating...
aws_launch_configuration.custom-launch-configuration: Creation complete after 2s [id=custom-launch-configuration]
aws_route_table_association.customvpc-public-2-rta: Creation complete after 1s [id=rtbassoc-038d6943a7eb40bd0]
aws_route_table_association.customvpc-public-1-rta: Creation complete after 1s [id=rtbassoc-03f562239013f9791]
aws_elb.custom-elb: Creation complete after 9s [id=custom-elb]
aws_autoscaling_group.custom-group-autoscaling: Creating...
aws_nat_gateway.customvpc-nat-gw: Still creating... [10s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [10s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [20s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [30s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [30s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [40s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [40s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [50s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [50s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m0s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [1m0s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m10s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still creating... [1m10s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Creation complete after 1m13s [id=custom-group-autoscaling]
aws_autoscaling_policy.custom-cpu-policy-scaledown: Creating...
aws_autoscaling_policy.custom-cpu-policy: Creating...
aws_autoscaling_policy.custom-cpu-policy-scaledown: Creation complete after 2s [id=custom-cpu-policy-scaledown]
aws_autoscaling_policy.custom-cpu-policy: Creation complete after 2s [id=custom-cpu-policy]
aws_nat_gateway.customvpc-nat-gw: Still creating... [1m30s elapsed]
aws_nat_gateway.customvpc-nat-gw: Creation complete after 1m39s [id=nat-037bb60f633bf4f1e]
aws_route_table.customvpc-private-rt: Creating...
aws_route_table.customvpc-private-rt: Creation complete after 3s [id=rtb-0d1ea2fbbf2ee4422]
aws_route_table_association.customvpc-private-rta: Creating...
aws_route_table_association.customvpc-private-rta: Creation complete after 1s [id=rtbassoc-0821300893c1787d4]

Apply complete! Resources: 19 added, 0 changed, 0 destroyed.

Outputs:

ELB = "custom-elb-38138711.us-east-1.elb.amazonaws.com"
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2#
```



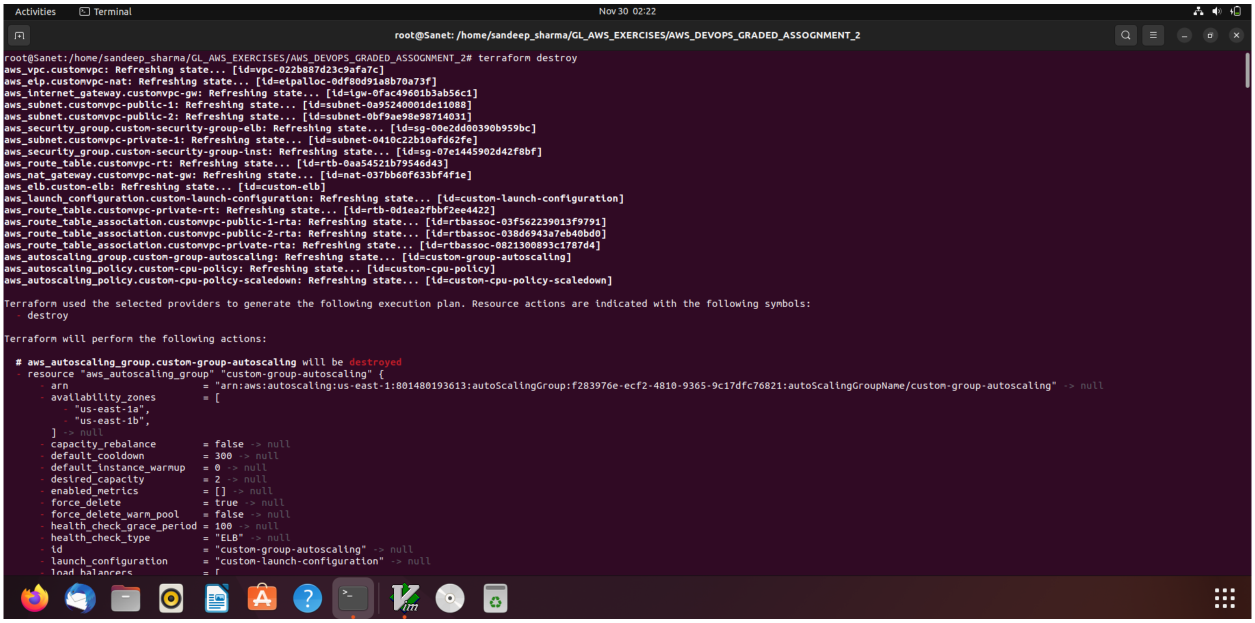
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

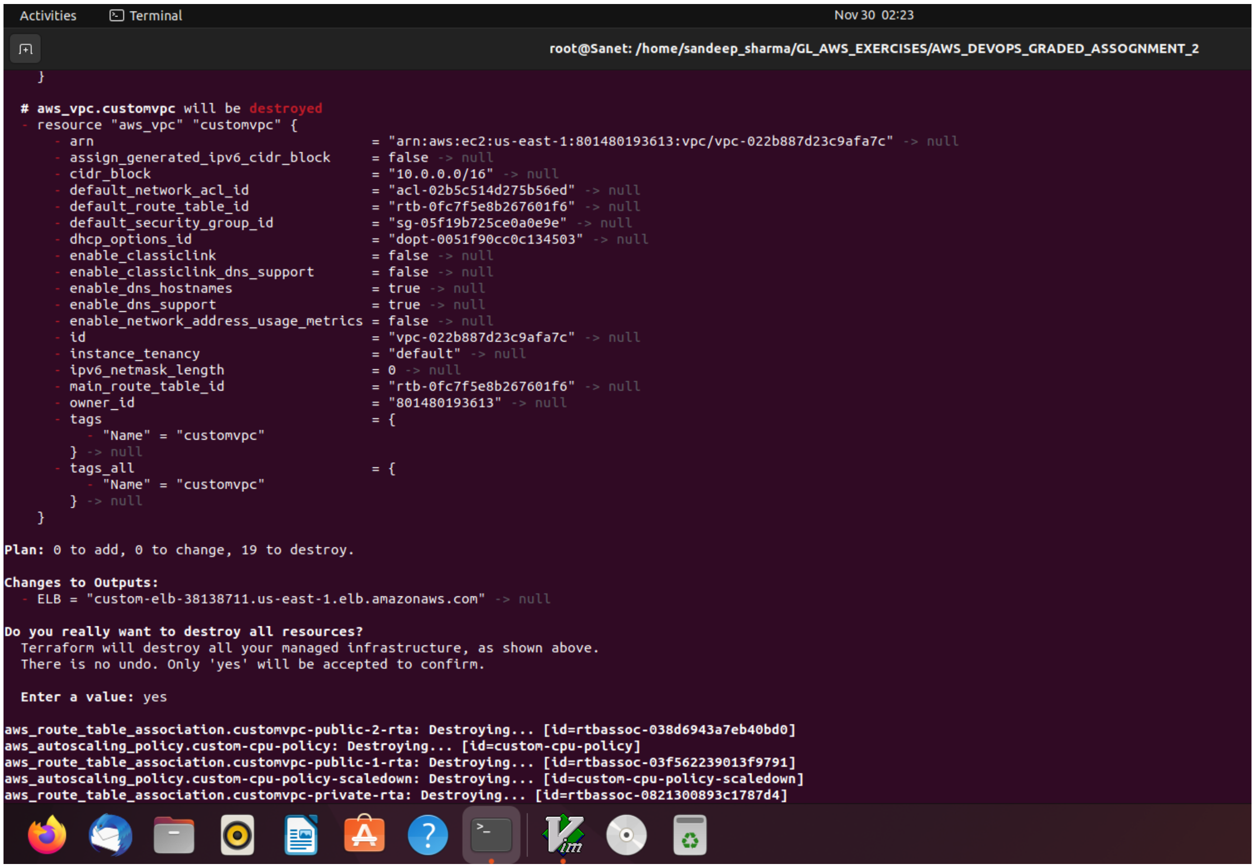
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

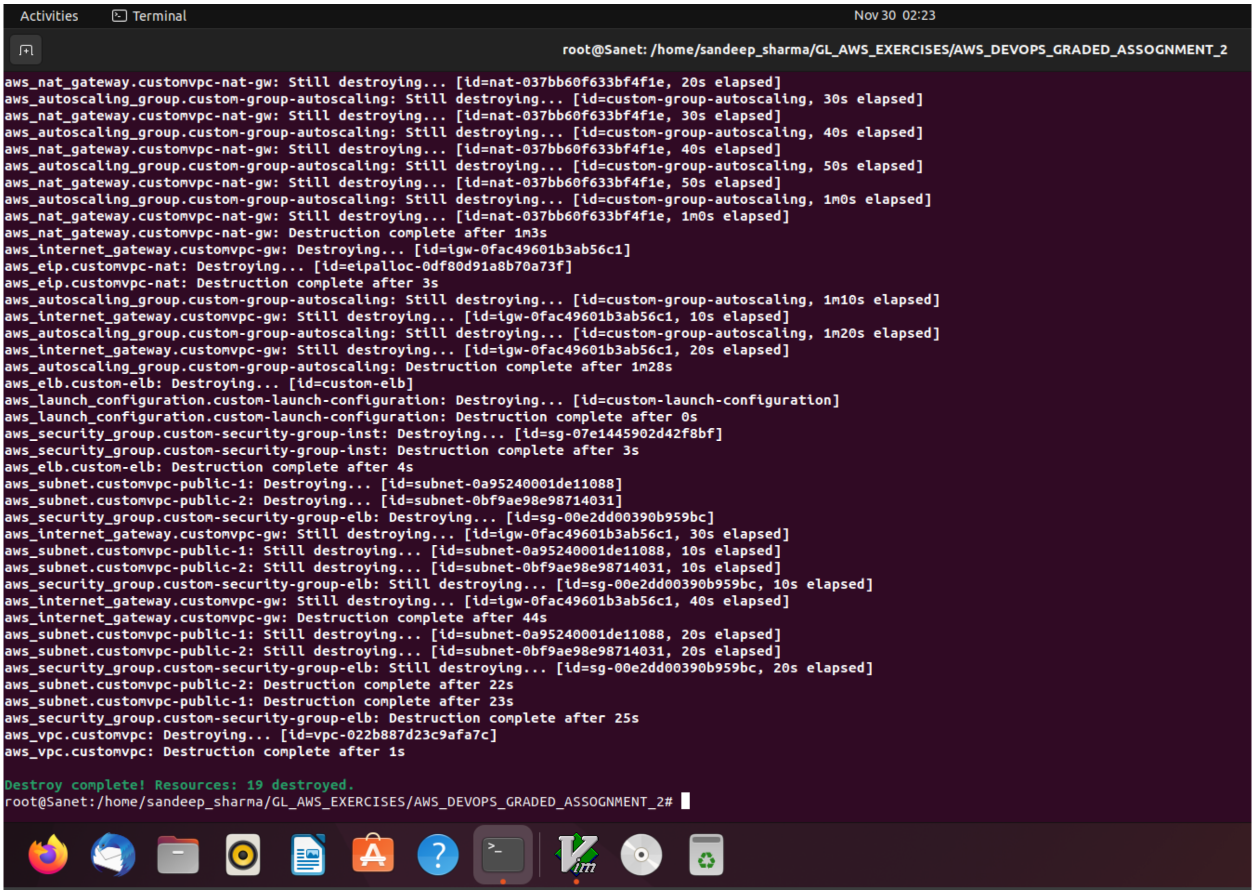
Destroying all procured resources:


Activities
Terminal
Nov 30 02:22
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# terraform destroy
aws_vpc.customvpc: Refreshing state... [id=vpc-022b887d23c9afa7c]
aws_eip.customvpc-nat: Refreshing state... [id=eipalloc-0df80d91a8b70a73f]
aws_internet_gateway.customvpc-gw: Refreshing state... [id=igw-0fac49601b3ab56c1]
aws_subnet.customvpc-public-1: Refreshing state... [id=subnet-0a95240001de11088]
aws_subnet.customvpc-public-2: Refreshing state... [id=subnet-0bf9ae98e98714031]
aws_security_group.custom-security-group-elb: Refreshing state... [id=sg-00e2dd00390b959bc]
aws_subnet.customvpc-private-1: Refreshing state... [id=subnet-0410c22b10afd62fe]
aws_security_group.custom-security-group-inst: Refreshing state... [id=sg-07e1445902d42f8bf]
aws_route_table.customvpc-rt: Refreshing state... [id=rtb-0aa54521b79546d43]
aws_nat_gateway.customvpc-nat-gw: Refreshing state... [id=nat-037bb60f633bf4f1e]
aws_elb.custom-elb: Refreshing state... [id=custom-elb]
aws_launch_configuration.custom-launch-configuration: Refreshing state... [id=custom-launch-configuration]
aws_route_table.customvpc-private-rt: Refreshing state... [id=rtb-0d1ea2fbbf2ee4422]
aws_route_table_association.customvpc-public-1-rta: Refreshing state... [id=rtbassoc-03f562239013f9791]
aws_route_table_association.customvpc-public-2-rta: Refreshing state... [id=rtbassoc-038d6943a7eb40bd0]
aws_route_table_association.customvpc-private-rta: Refreshing state... [id=rtbassoc-0821300893c1787d4]
aws_autoscaling_group.custom-group-autoscaling: Refreshing state... [id=custom-group-autoscaling]
aws_autoscaling_policy.custom-cpu-policy: Refreshing state... [id=custom-cpu-policy]
aws_autoscaling_policy.custom-cpu-policy-scaledown: Refreshing state... [id=custom-cpu-policy-scaledown]

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

Terraform will perform the following actions:

  # aws_autoscaling_group.custom-group-autoscaling will be destroyed
  - resource "aws_autoscaling_group" "custom-group-autoscaling" {
      - arn                       = "arn:aws:autoscaling:us-east-1:801480193613:autoScalingGroup:f283976e-ecf2-4810-9365-9c17dfc76821:autoScalingGroupName/custom-group-autoscaling" -> null
      - availability_zones        = [
          - "us-east-1a",
          - "us-east-1b",
        ] -> null
      - capacity_rebalance        = false -> null
      - default_cooldown          = 300 -> null
      - default_instance_warmup   = 0 -> null
      - desired_capacity          = 2 -> null
      - enabled_metrics           = [] -> null
      - force_delete              = true -> null
      - force_delete_warm_pool    = false -> null
      - health_check_grace_period = 100 -> null
      - health_check_type         = "ELB" -> null
      - id                        = "custom-group-autoscaling" -> null
      - launch_configuration      = "custom-launch-configuration" -> null
      - load_balancers            = [
```


Activities
Terminal
Nov 30 02:23
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
    }

  # aws_vpc.customvpc will be destroyed
  - resource "aws_vpc" "customvpc" {
      - arn                                  = "arn:aws:ec2:us-east-1:801480193613:vpc/vpc-022b887d23c9afa7c" -> null
      - assign_generated_ipv6_cidr_block     = false -> null
      - cidr_block                           = "10.0.0.0/16" -> null
      - default_network_acl_id               = "acl-02b5c514d275b56ed" -> null
      - default_route_table_id               = "rtb-0fc7f5e8b267601f6" -> null
      - default_security_group_id            = "sg-05f19b725ce0a0e9e" -> null
      - dhcp_options_id                      = "dopt-0051f90cc0c134503" -> null
      - enable_classiclink                   = false -> null
      - enable_classiclink_dns_support       = false -> null
      - enable_dns_hostnames                 = true -> null
      - enable_dns_support                   = true -> null
      - enable_network_address_usage_metrics = false -> null
      - id                                   = "vpc-022b887d23c9afa7c" -> null
      - instance_tenancy                     = "default" -> null
      - ipv6_netmask_length                  = 0 -> null
      - main_route_table_id                  = "rtb-0fc7f5e8b267601f6" -> null
      - owner_id                             = "801480193613" -> null
      - tags                                 = {
          - "Name" = "customvpc"
        } -> null
      - tags_all                             = {
          - "Name" = "customvpc"
        } -> null
    }

Plan: 0 to add, 0 to change, 19 to destroy.

Changes to Outputs:
  - ELB = "custom-elb-38138711.us-east-1.elb.amazonaws.com" -> null

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_route_table_association.customvpc-public-2-rta: Destroying... [id=rtbassoc-038d6943a7eb40bd0]
aws_autoscaling_policy.custom-cpu-policy: Destroying... [id=custom-cpu-policy]
aws_route_table_association.customvpc-public-1-rta: Destroying... [id=rtbassoc-03f562239013f9791]
aws_autoscaling_policy.custom-cpu-policy-scaledown: Destroying... [id=custom-cpu-policy-scaledown]
aws_route_table_association.customvpc-private-rta: Destroying... [id=rtbassoc-0821300893c1787d4]
```


Activities
Terminal
Nov 30 02:23
root@Sanet: /home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2

```
aws_nat_gateway.customvpc-nat-gw: Still destroying... [id=nat-037bb60f633bf4f1e, 20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 30s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still destroying... [id=nat-037bb60f633bf4f1e, 30s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 40s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still destroying... [id=nat-037bb60f633bf4f1e, 40s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 50s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still destroying... [id=nat-037bb60f633bf4f1e, 50s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 1m0s elapsed]
aws_nat_gateway.customvpc-nat-gw: Still destroying... [id=nat-037bb60f633bf4f1e, 1m0s elapsed]
aws_nat_gateway.customvpc-nat-gw: Destruction complete after 1m3s
aws_internet_gateway.customvpc-gw: Destroying... [id=igw-0fac49601b3ab56c1]
aws_eip.customvpc-nat: Destroying... [id=eipalloc-0df80d91a8b70a73f]
aws_eip.customvpc-nat: Destruction complete after 3s
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 1m10s elapsed]
aws_internet_gateway.customvpc-gw: Still destroying... [id=igw-0fac49601b3ab56c1, 10s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Still destroying... [id=custom-group-autoscaling, 1m20s elapsed]
aws_internet_gateway.customvpc-gw: Still destroying... [id=igw-0fac49601b3ab56c1, 20s elapsed]
aws_autoscaling_group.custom-group-autoscaling: Destruction complete after 1m28s
aws_elb.custom-elb: Destroying... [id=custom-elb]
aws_launch_configuration.custom-launch-configuration: Destroying... [id=custom-launch-configuration]
aws_launch_configuration.custom-launch-configuration: Destruction complete after 0s
aws_security_group.custom-security-group-inst: Destroying... [id=sg-07e1445902d42f8bf]
aws_security_group.custom-security-group-inst: Destruction complete after 3s
aws_elb.custom-elb: Destruction complete after 4s
aws_subnet.customvpc-public-1: Destroying... [id=subnet-0a95240001de11088]
aws_subnet.customvpc-public-2: Destroying... [id=subnet-0bf9ae98e98714031]
aws_security_group.custom-security-group-elb: Destroying... [id=sg-00e2dd00390b959bc]
aws_internet_gateway.customvpc-gw: Still destroying... [id=igw-0fac49601b3ab56c1, 30s elapsed]
aws_subnet.customvpc-public-1: Still destroying... [id=subnet-0a95240001de11088, 10s elapsed]
aws_subnet.customvpc-public-2: Still destroying... [id=subnet-0bf9ae98e98714031, 10s elapsed]
aws_security_group.custom-security-group-elb: Still destroying... [id=sg-00e2dd00390b959bc, 10s elapsed]
aws_internet_gateway.customvpc-gw: Still destroying... [id=igw-0fac49601b3ab56c1, 40s elapsed]
aws_internet_gateway.customvpc-gw: Destruction complete after 44s
aws_subnet.customvpc-public-1: Still destroying... [id=subnet-0a95240001de11088, 20s elapsed]
aws_subnet.customvpc-public-2: Still destroying... [id=subnet-0bf9ae98e98714031, 20s elapsed]
aws_security_group.custom-security-group-elb: Still destroying... [id=sg-00e2dd00390b959bc, 20s elapsed]
aws_subnet.customvpc-public-2: Destruction complete after 22s
aws_subnet.customvpc-public-1: Destruction complete after 23s
aws_security_group.custom-security-group-elb: Destruction complete after 25s
aws_vpc.customvpc: Destroying... [id=vpc-022b887d23c9afa7c]
aws_vpc.customvpc: Destruction complete after 1s

Destroy complete! Resources: 19 destroyed.
root@Sanet:/home/sandeep_sharma/GL_AWS_EXERCISES/AWS_DEVOPS_GRADED_ASSOGNMENT_2# 
```

Codes uploaded to github also:

https://github.com/Sandeep-Sharma-3/Sandeep_Sharma_Cloud_Computing_Graded_Assignment_2.git