Fatima Jinnah Women University

Subject: Cloud Computing

Lab 12

Name:

Tehreem khan(5-B)

Registration number:

2023-BSE-064

Submitted To:

Sir Shoaib

Task 0:

```
PS C:\Users\tehre> gh codespace list
NAME                                 DISPLAY NAME        REPOSITORY                                  BRANCH  STATE      CREATED AT
symmetrical-cod-9664p55vpjph95gv     symmetrical cod     tehreem-0514/CC_TehreemKhan_064             main    Available  about 1 day ago
didactic-waddle-9664p55vpx4c7r56     didactic waddle     tehreem-0514/CC_TehreemKhan_064_Lab11       main*   Shutdown   about 1 day ago
refactored-goggles-wrq764v4v54vf9vrp refactored goggles tehreem-0514/CC_TehreemKhan_064_Lab12       main    Available  about 3 minutes ago
PS C:\Users\tehre>
```

```
PS C:\Users\tehre> gh codespace ssh -c refactored-goggles-wrq764v4v54vf9vrp
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

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.

@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab12 (main) $
```

Task 1:

```
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab12 (main) $ mkdir -p ~/Lab12
@tehreem-0514 ➜ /workspaces/CC_TehreemKhan_064_Lab12 (main) $ cd ~/Lab12
@tehreem-0514 ➜ ~/Lab12 $
```

```
@tehreem-0514 ➜ ~/Lab12 $ touch main.tf variables.tf outputs.tf locals.tf terraform.tfvars entry-script.sh
@tehreem-0514 ➜ ~/Lab12 $ ls -la
total 12
drwxrwxr-x 2 codespace codespace 4096 Jan  3 18:30 .
drwxr-x--- 1 codespace codespace 4096 Jan  3 18:29 ..
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 entry-script.sh
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 locals.tf
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 main.tf
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 outputs.tf
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 terraform.tfvars
-rw-rw-r-- 1 codespace codespace    0 Jan  3 18:30 variables.tf
@tehreem-0514 ➜ ~/Lab12 $
```

```
  GNU nano 7.2                                variables.tf *
variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
variable "instance_type" {}
variable "public_key" {}
variable "private_key" {}
```

```
  GNU nano 7.2                                 outputs.tf *
output "aws_instance_public_ip" {
  value = aws_instance.myapp-server.public_ip
}
```

GNU nano 7.2 locals.tf *

```
locals {
  my_ip = "${chomp(data.http.my_ip.response_body)}/32"
}

data "http" "my_ip" {
  url = "https://icanhazip.com"
}
```

GNU nano 7.2 terraform.tfvars *

```
vpc_cidr_block = "10.0.0.0/16"
subnet_cidr_block = "10.0.10.0/24"
availability_zone = "me-central-1a"
env_prefix = "dev"
instance_type = "t3.micro"
public_key = "~/.ssh/id_ed25519.pub"
private_key = "~/.ssh/id_ed25519"
```

GNU nano 7.2 main.tf *

```
    protocol    = "tcp"
    cidr_blocks = [local.my_ip]
  }

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

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

  tags = {
     Name = "${var.env_prefix}-default-sg"
  }
}

resource "aws_key_pair" "ssh-key" {
  key_name   = "serverkey"
  public_key = file(var.public_key)
}

resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6"
  instance_type = var.instance_type
  subnet_id     = aws_subnet.myapp_subnet_1.id
  security_groups = [aws_default_security_group.default_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

  user_data = file("./entry-script.sh")

  tags = {
     Name = "${var.env_prefix}-ec2-instance"
  }
}
```

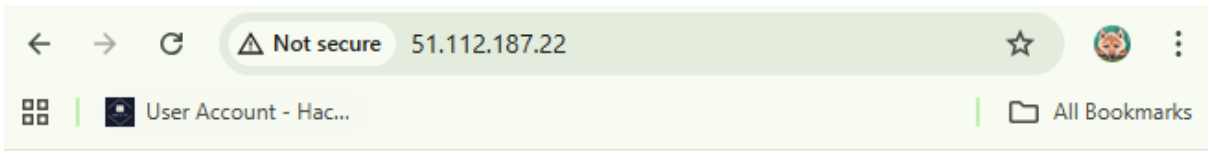
```
@tehreem-0514 ~/Lab12 $ nano entry-script.sh
@tehreem-0514 ~/Lab12 $ @tehreem-0514 ~/Lab12 $ chmod +x entry-script.sh
@tehreem-0514 ~/Lab12 $ cat entry-script.sh
#!/bin/bash
set -e
yum update -y
yum install -y nginx
systemctl start nginx
systemctl enable nginx

@tehreem-0514 ~/Lab12 $
```

```
@tehreem-0514 ~/Lab12 $ ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -N ""
Generating public/private ed25519 key pair.
Your identification has been saved in /home/codespace/.ssh/id_ed25519
Your public key has been saved in /home/codespace/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:[redacted]
The key's randomart image is:
+--[ED25519 256]--+
|         o= oBBo |
|        .=.o+.oo |
|        .oo. +o  |
|         .o +... |
|        S .o ++o.|
|          .. .+*.o|
|            +.+.o=|
|             o.++E|
|              .=X+|
+----[SHA256]-----+
@tehreem-0514 ~/Lab12 $
```

```
@tehreem-0514 ~/Lab12 $ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/http...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/http v3.5.0...
- Installed hashicorp/http v3.5.0 (signed by HashiCorp)
- Installing hashicorp/aws v6.27.0...
- Installed hashicorp/aws v6.27.0 (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.
@tehreem-0514 ~/Lab12 $
```

```
Changes to Outputs:
  + aws_instance_public_ip = (known after apply)
aws_key_pair.ssh-key: Creating...
aws_vpc.myapp_vpc: Creating...
aws_key_pair.ssh-key: Creation complete after 1s [id=serverkey]
aws_vpc.myapp_vpc: Creation complete after 2s [id=vpc-0ea76a19a8a366fd4]
aws_internet_gateway.myapp_igw: Creating...
aws_subnet.myapp_subnet_1: Creating...
aws_default_security_group.default_sg: Creating...
aws_internet_gateway.myapp_igw: Creation complete after 0s [id=igw-01c698c1d9fbafc50]
aws_default_route_table.main_rt: Creating...
aws_subnet.myapp_subnet_1: Creation complete after 1s [id=subnet-080169e145b0d891f]
aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-0294fbf6ad21f1c31]
aws_default_security_group.default_sg: Creation complete after 2s [id=sg-02354cffc9d40e326]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-0ec55f115633f8376]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

aws_instance_public_ip = "51.112.187.22"
@tehreem-0514 ▣ ~/Lab12 $
```

```
@tehreem-0514 ▣ ~/Lab12 $
@tehreem-0514 ▣ ~/Lab12 $ terraform output
aws_instance_public_ip = "51.112.187.22"
@tehreem-0514 ▣ ~/Lab12 $
```



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.

Task 2:

```
 GNU nano 7.2                                    main.tf *
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
    prefix_list_ids = []
  }

  tags = {
     Name = "${var.env_prefix}-default-sg"
  }
}

resource "aws_key_pair" "ssh-key" {
  key_name   = "serverkey"
  public_key = file(var.public_key)
}

resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6"
  instance_type = var.instance_type
  subnet_id     = aws_subnet.myapp_subnet_1.id
  security_groups = [aws_default_security_group.default_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

  connection {
    type        = "ssh"
    user        = "ec2-user"
    private_key = file(var.private_key)
    host        = self.public_ip
  }

  provisioner "remote-exec" {
    inline = [
      "sudo yum update -y",
      "sudo yum install -y nginx",
      "sudo systemctl start nginx",
      "sudo systemctl enable nginx"
    ]
  }

  tags = {
    Name = "${var.env_prefix}-ec2-instance"
  }
}
```

```
aws_instance.myapp-server (remote-exec):   Installing       : gperf [====   ] 4/7
aws_instance.myapp-server (remote-exec):   Installing       : gperf [===== ] 4/7
aws_instance.myapp-server (remote-exec):   Installing       : gperftools-l   4/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [      ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [=     ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [==    ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [===   ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [====  ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [===== ] 5/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx-core-1   5/7
aws_instance.myapp-server (remote-exec):   Installing       : gener [      ] 6/7
aws_instance.myapp-server (remote-exec):   Installing       : gener [====  ] 6/7
aws_instance.myapp-server (remote-exec):   Installing       : gener [===== ] 6/7
aws_instance.myapp-server (remote-exec):   Installing       : generic-logo   6/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [      ] 7/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [==    ] 7/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [===   ] 7/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx [===== ] 7/7
aws_instance.myapp-server (remote-exec):   Installing       : nginx-1:1.28   7/7
aws_instance.myapp-server (remote-exec):   Running scriptlet: nginx-1:1.28   7/7
aws_instance.myapp-server: Still creating... [00m30s elapsed]
aws_instance.myapp-server (remote-exec):   Verifying        : generic-logo   1/7
aws_instance.myapp-server (remote-exec):   Verifying        : gperftools-l   2/7
aws_instance.myapp-server (remote-exec):   Verifying        : libunwind-1.   3/7
aws_instance.myapp-server (remote-exec):   Verifying        : nginx-1:1.28   4/7
aws_instance.myapp-server (remote-exec):   Verifying        : nginx-core-1   5/7
aws_instance.myapp-server (remote-exec):   Verifying        : nginx-filesy   6/7
aws_instance.myapp-server (remote-exec):   Verifying        : nginx-mimety   7/7

aws_instance.myapp-server (remote-exec): Installed:
aws_instance.myapp-server (remote-exec):   generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
aws_instance.myapp-server (remote-exec):   gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64
aws_instance.myapp-server (remote-exec):   libunwind-1.4.0-5.amzn2023.0.3.x86_64
aws_instance.myapp-server (remote-exec):   nginx-1:1.28.0-1.amzn2023.0.2.x86_64
aws_instance.myapp-server (remote-exec):   nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64
aws_instance.myapp-server (remote-exec):   nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch
aws_instance.myapp-server (remote-exec):   nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch

aws_instance.myapp-server (remote-exec): Complete!
aws_instance.myapp-server (remote-exec): Created symlink /etc/systemd/system/multi-user.target.want
s/nginx.service → /usr/lib/systemd/system/nginx.service.
aws_instance.myapp-server: Creation complete after 31s [id=i-0cfa5140eb03887bf]

Apply complete! Resources: 1 added, 1 changed, 1 destroyed.

Outputs:

aws_instance_public_ip = "3.28.132.103"
@tehreem-0514 ▨ ~/Lab12 $
```

```
@tehreem-0514 ▨ ~/Lab12 $ terraform output
aws_instance_public_ip = "3.28.132.103"
@tehreem-0514 ▨ ~/Lab12 $
```

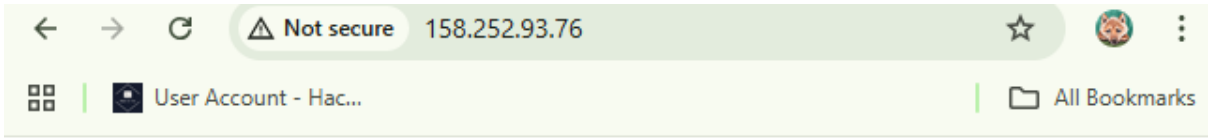

Not secure
3.28.132.103
User Account - Hac...
All Bookmarks
Welcome to nginx!
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.
Thank you for using nginx.

Task 3:

```
  GNU nano 7.2                                  main.tf *
}

resource "aws_key_pair" "ssh-key" {
  key_name   = "serverkey"
  public_key = file(var.public_key)
}
resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6"
  instance_type = var.instance_type
  subnet_id     = aws_subnet.myapp_subnet_1.id
  security_groups = [aws_default_security_group.default_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

  connection {
    type        = "ssh"
    user        = "ec2-user"
    private_key = file(var.private_key)
    host        = self.public_ip
  }

  provisioner "file" {
    source      = "./entry-script.sh"
    destination = "/home/ec2-user/entry-script-on-ec2.sh"
  }

  provisioner "remote-exec" {
    inline = [
      "sudo chmod +x /home/ec2-user/entry-script-on-ec2.sh",
      "sudo /home/ec2-user/entry-script-on-ec2.sh"
    ]
  }

  provisioner "local-exec" {
    command = <<-EOF
      echo Instance ${self.id} with public IP ${self.public_ip} has been created
    EOF
  }

  tags = {
    Name = "${var.env_prefix}-ec2-instance"
  }
}
```

```
aws_instance.myapp-server (remote-exec): Complete!
aws_instance.myapp-server (remote-exec): Created symlink /etc/systemd/system/multi-user.target.want
s/nginx.service → /usr/lib/systemd/system/nginx.service.
aws_instance.myapp-server: Provisioning with 'local-exec'...
aws_instance.myapp-server (local-exec): Executing: ["/bin/sh" "-c" "echo Instance i-02a5ea07339f7a8
19 with public IP 158.252.93.76 has been created\n"]
aws_instance.myapp-server (local-exec): Instance i-02a5ea07339f7a819 with public IP 158.252.93.76 h
as been created
aws_instance.myapp-server: Creation complete after 33s [id=i-02a5ea07339f7a819]

Apply complete! Resources: 1 added, 0 changed, 1 destroyed.

Outputs:

aws_instance_public_ip = "158.252.93.76"
```

```
@tehreem-0514 ~/Lab12 $ terraform output
aws_instance_public_ip = "158.252.93.76"
@tehreem-0514 ~/Lab12 $
```



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.

```
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_default_route_table.main_rt: Destroying... [id=rtb-0294fbf6ad21f1c31]
aws_instance.myapp-server: Destroying... [id=i-02a5ea07339f7a819]
aws_default_route_table.main_rt: Destruction complete after 0s
aws_internet_gateway.myapp_igw: Destroying... [id=igw-01c698c1d9fbafc50]
aws_instance.myapp-server: Still destroying... [id=i-02a5ea07339f7a819, 00m10s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-01c698c1d9fbafc50, 00m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-02a5ea07339f7a819, 00m20s elapsed]
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-01c698c1d9fbafc50, 00m20s elapsed]
aws_internet_gateway.myapp_igw: Destruction complete after 27s
aws_instance.myapp-server: Still destroying... [id=i-02a5ea07339f7a819, 00m30s elapsed]
aws_instance.myapp-server: Destruction complete after 30s
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_subnet.myapp_subnet_1: Destroying... [id=subnet-080169e145b0d891f]
aws_default_security_group.default_sg: Destroying... [id=sg-02354cffc9d40e326]
aws_default_security_group.default_sg: Destruction complete after 0s
aws_key_pair.ssh-key: Destruction complete after 1s
aws_subnet.myapp_subnet_1: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-0ea76a19a8a366fd4]
aws_vpc.myapp_vpc: Destruction complete after 0s

Destroy complete! Resources: 7 destroyed.
@tehreem-0514 ~/Lab12 $
```

```
  GNU nano 7.2                                  main.tf *

resource "aws_key_pair" "ssh-key" {
  key_name   = "serverkey"
  public_key = file(var.public_key)
}
resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6"
  instance_type = var.instance_type
  subnet_id     = aws_subnet.myapp_subnet_1.id
  security_groups = [aws_default_security_group.default_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

user_data = file("./entry-script.sh")
```

Task 4:

```
@tehreem-0514 ▨ ~/Lab12 $ @tehreem-0514 ▨ ~/Lab12 $ mkdir -p modules/subnet
t/variab@tehreem-0514 ▨ ~/Lab12 $ touch modules/subnet/main.tf
@tehreem-0514 ▨ ~/Lab12 $ touch modules/subnet/variables.tf
@tehreem-0514 ▨ ~/Lab12 $ touch modules/subnet/outputs.tf
@tehreem-0514 ▨ ~/Lab12 $ tree modules
modules
└── subnet
    ├── main.tf
    ├── outputs.tf
    ├── outputs.tftouch
    └── variables.tf

2 directories, 4 files
@tehreem-0514 ▨ ~/Lab12 $
```

```
 GNU nano 7.2                          modules/subnet/variables.tf *
variable "vpc_id" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
variable "default_route_table_id" {}
```

```
 GNU nano 7.2                            modules/subnet/main.tf *
resource "aws_subnet" "myapp_subnet_1" {
  vpc_id                  = var.vpc_id
  cidr_block              = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  map_public_ip_on_launch = true

  tags = {
    Name = "${var.env_prefix}-subnet-1"
  }
}

resource "aws_internet_gateway" "myapp_igw" {
  vpc_id = var.vpc_id

  tags = {
    Name = "${var.env_prefix}-igw"
  }
}

resource "aws_default_route_table" "main_rt" {
  default_route_table_id = var.default_route_table_id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.myapp_igw.id
  }

  tags = {
    Name = "${var.env_prefix}-rt"
  }
}
```

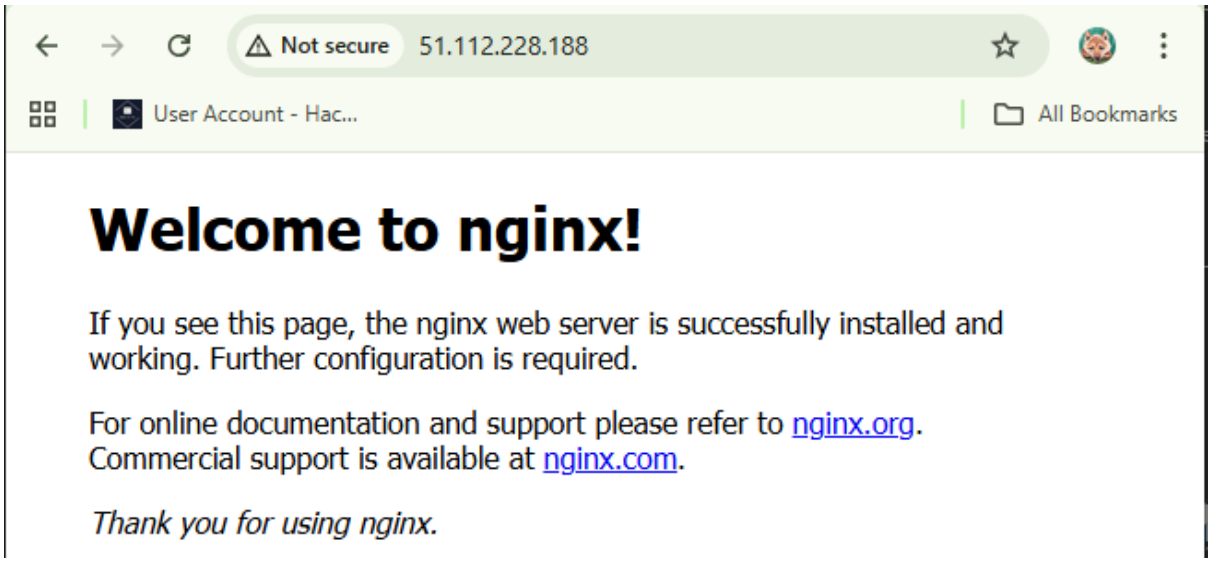
```
 GNU nano 7.2                          modules/subnet/outputs.tf *
output "subnet" {
  value = aws_subnet.myapp_subnet_1
}
```

```
 GNU nano 7.2                                  main.tf *
    protocol    = "tcp"
    cidr_blocks = [local.my_ip]
  }

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

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

  tags = {
     Name = "${var.env_prefix}-default-sg"
  }
}

resource "aws_key_pair" "ssh-key" {
  key_name   = "serverkey"
  public_key = file(var.public_key)
}

resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6"
  instance_type               = var.instance_type
  subnet_id                   = module.myapp-subnet.subnet.id
  security_groups             = [aws_default_security_group.default_sg.id]
  availability_zone           = var.availability_zone
  associate_public_ip_address = true
  key_name                    = aws_key_pair.ssh-key.key_name

  user_data = file("./entry-script.sh")

  tags = {
    Name = "${var.env_prefix}-ec2-instance"
  }
}
```

```
@tehreem-0514 ~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-subnet in modules/subnet
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/http from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/http v3.5.0

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.
@tehreem-0514 ~/Lab12 $
```

```
Plan: 7 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_instance_public_ip = (known after apply)
aws_key_pair.ssh-key: Creating...
aws_vpc.myapp_vpc: Creating...
aws_key_pair.ssh-key: Creation complete after 0s [id=serverkey]
aws_vpc.myapp_vpc: Creation complete after 2s [id=vpc-09b19f409db216ea1]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creating...
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creating...
aws_default_security_group.default_sg: Creating...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creation complete after 0s [id=igw-0c443ad23bbc
f44ed]
module.myapp-subnet.aws_default_route_table.main_rt: Creating...
module.myapp-subnet.aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-0ddf801458c
e7b8fd]
aws_default_security_group.default_sg: Creation complete after 2s [id=sg-0e338b99c693f2fdd]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Still creating... [00m10s elapsed]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creation complete after 11s [id=subnet-08b76bd2547ca
83bd]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 12s [id=i-0401f40ebe1c19594]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

aws_instance_public_ip = "51.112.228.188"
@tehreem-0514 ▨ ~/Lab12 $
```

```
@tehreem-0514 ▨ ~/Lab12 $ terraform output
aws_instance_public_ip = "51.112.228.188"
@tehreem-0514 ▨ ~/Lab12 $
```



Task 5:

```
@tehreem-0514 ~/Lab12 $ mkdir -p modules/webserver
@tehreem-0514 ~/Lab12 $ touch modules/webserver/main.tf
@tehreem-0514 ~/Lab12 $ touch modules/webserver/variables.tf
@tehreem-0514 ~/Lab12 $ touch modules/webserver/outputs.tf
@tehreem-0514 ~/Lab12 $
```

```
  GNU nano 7.2                          modules/webserver/variables.tf *
variable "env_prefix" {}
variable "instance_type" {}
variable "availability_zone" {}
variable "public_key" {}
variable "my_ip" {}
variable "vpc_id" {}
variable "subnet_id" {}
variable "script_path" {}
variable "instance_suffix" {}
```

```
  GNU nano 7.2                          modules/webserver/main.tf *
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

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

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

  tags = {
    Name = "${var.env_prefix}-web-sg-${var.instance_suffix}"
  }
}

resource "aws_key_pair" "ssh-key" {
  key_name   = "${var.env_prefix}-serverkey-${var.instance_suffix}"
  public_key = file(var.public_key)
}

resource "aws_instance" "myapp-server" {
  ami                         = "ami-05524d6658fcf35b6"
  instance_type               = var.instance_type
  subnet_id                   = var.subnet_id
  vpc_security_group_ids      = [aws_security_group.web_sg.id]
  availability_zone           = var.availability_zone
  associate_public_ip_address = true
  key_name                    = aws_key_pair.ssh-key.key_name

  user_data = file(var.script_path)

  tags = {
    Name = "${var.env_prefix}-ec2-instance-${var.instance_suffix}"
  }
}
```

```
  GNU nano 7.2                          modules/webserver/outputs.tf *
output "aws_instance" {
  value = aws_instance.myapp-server
}
```

```
  GNU nano 7.2                                  main.tf *
  }
}

module "myapp-subnet" {
  source = "./modules/subnet"

  vpc_id                  = aws_vpc.myapp_vpc.id
  subnet_cidr_block       = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  env_prefix              = var.env_prefix
  default_route_table_id  = aws_vpc.myapp_vpc.default_route_table_id
}

module "myapp-webserver" {
  source            = "./modules/webserver"
  env_prefix        = var.env_prefix
  instance_type     = var.instance_type
  availability_zone = var.availability_zone
  public_key        = var.public_key
  my_ip             = local.my_ip
  vpc_id            = aws_vpc.myapp_vpc.id
  subnet_id         = module.myapp-subnet.subnet.id
  script_path       = "./entry-script.sh"
  instance_suffix   = "0"
}
```

```
  GNU nano 7.2                                  outputs.tf *
output "webserver_public_ip" {
  value = module.myapp-webserver.aws_instance.public_ip
}
```

```
@tehreem-0514 ~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-webserver in modules/webserver
Initializing provider plugins...
- Reusing previous version of hashicorp/http from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/http v3.5.0
- Using previously-installed hashicorp/aws v6.27.0

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.
@tehreem-0514 ~/Lab12 $
```

```
Plan: 3 to add, 0 to change, 3 to destroy.

Changes to Outputs:
  - aws_instance_public_ip = "51.112.228.188" -> null
  + webserver_public_ip    = (known after apply)
module.myapp-webserver.aws_key_pair.ssh-key: Creating...
aws_instance.myapp-server: Destroying... [id=i-0401f40ebe1c19594]
module.myapp-webserver.aws_security_group.web_sg: Creating...
module.myapp-webserver.aws_key_pair.ssh-key: Creation complete after 1s [id=dev-serverkey-0]
module.myapp-webserver.aws_security_group.web_sg: Creation complete after 3s [id=sg-0122c556f207952
f1]
module.myapp-webserver.aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Creation complete after 13s [id=i-0a0f2fdfc3a2893
8e]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 00m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 00m30s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 00m40s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 00m50s elapsed]

aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 01m00s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 01m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 01m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-0401f40ebe1c19594, 01m30s elapsed]
aws_instance.myapp-server: Destruction complete after 1m31s
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_default_security_group.default_sg: Destroying... [id=sg-0e338b99c693f2fdd]
aws_default_security_group.default_sg: Destruction complete after 0s
aws_key_pair.ssh-key: Destruction complete after 0s

Apply complete! Resources: 3 added, 0 changed, 3 destroyed.

Outputs:

webserver_public_ip = "51.112.50.204"
@tehreem-0514 ▨ ~/Lab12 $
@tehreem-0514 ▨ ~/Lab12 $
```

```
@tehreem-0514 ▨ ~/Lab12 $ terraform output
webserver_public_ip = "51.112.50.204"
@tehreem-0514 ▨ ~/Lab12 $
```

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.

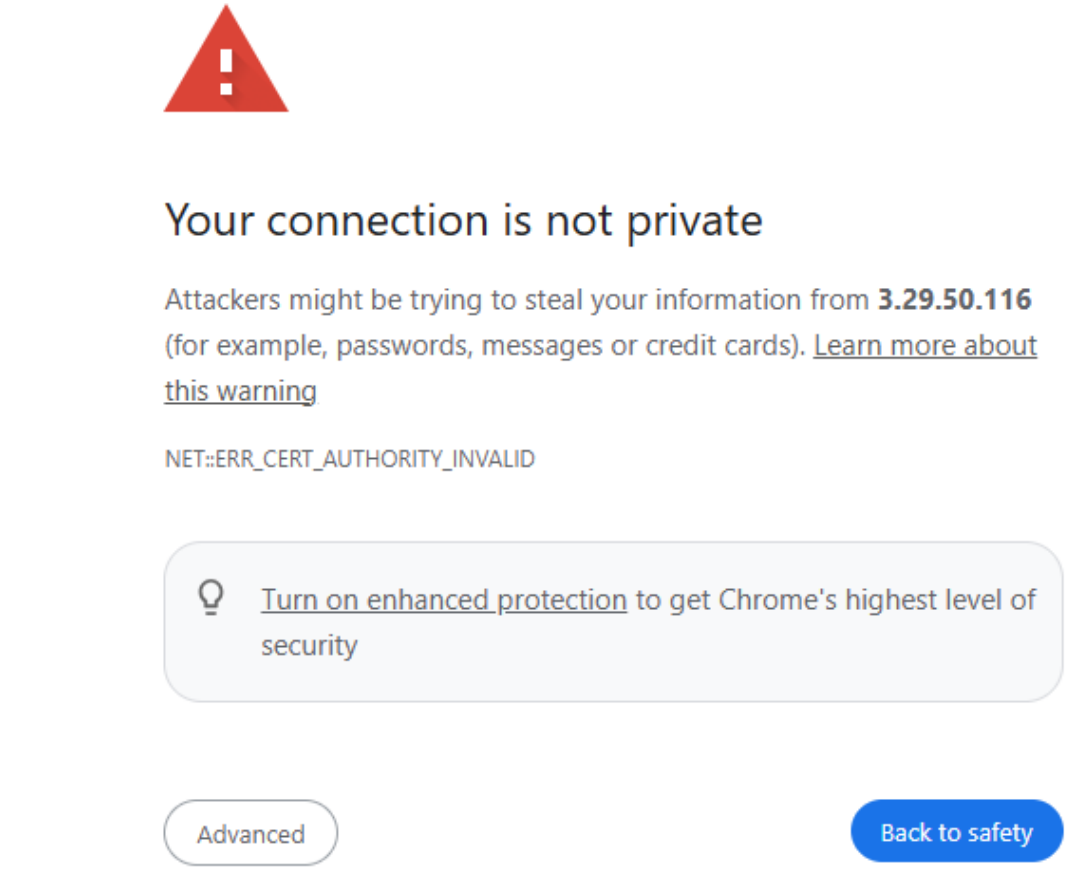
```
  - webserver_public_ip = "51.112.50.204" -> null

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

module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-0ddf801458ce7b8fd]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-webserver.aws_instance.myapp-server: Destroying... [id=i-0a0f2fdfc3a28938e]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id=igw-0c443ad23bbcf44ed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 00m1
0s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0c443ad23bbcf44ed,
00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 00m2
0s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0c443ad23bbcf44ed,
00m20s elapsed]

module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 00m3
0s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0c443ad23bbcf44ed,
00m30s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 00m4
0s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0c443ad23bbcf44ed,
00m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 00m5
0s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0c443ad23bbcf44ed,
00m50s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 58s
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a0f2fdfc3a28938e, 01m0
0s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Destruction complete after 1m1s
module.myapp-webserver.aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-0]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id=subnet-08b76bd2547ca83bd]
module.myapp-webserver.aws_security_group.web_sg: Destroying... [id=sg-0122c556f207952f1]
module.myapp-webserver.aws_key_pair.ssh-key: Destruction complete after 0s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 0s
module.myapp-webserver.aws_security_group.web_sg: Destruction complete after 0s
aws_vpc.myapp_vpc: Destroying... [id=vpc-09b19f409db216ea1]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@tehreem-0514 ~/Lab12 $
@tehreem-0514 ~/Lab12 $
```

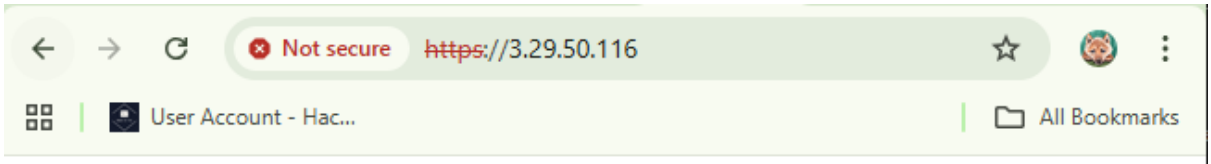
Task 6:

```
 GNU nano 7.2                                entry-script.sh *
http {
    log_format  main  '\$remote_addr - \$remote_user [\$time_local] "\$request"'
                      '\$status \$body_bytes_sent "\$http_referer"'
                      '"\$http_user_agent" "\$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile            on;
    tcp_nopush          on;
    keepalive_timeout   65;
    types_hash_max_size 4096;

    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    upstream backend_servers {
        server 158.252.94.241:80;
        server 158.252.94.242:80 backup;
    }

    server {
        listen 443 ssl;
        server_name $PUBLIC_IP;
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

        location / {
            root /usr/share/nginx/html;
            index index.html;
    #       proxy_pass http://158.252.94.241:80;
    #       proxy_pass http://backend_servers;

        }
    }

    server {
        listen 80;
        server_name _;
        return 301 https://\$host\$request_uri;
    }
}
EOF

# Test and restart Nginx
systemctl restart nginx
```

```
Plan: 7 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + webserver_public_ip = (known after apply)
module.myapp-webserver.aws_key_pair.ssh-key: Creating...
aws_vpc.myapp_vpc: Creating...
module.myapp-webserver.aws_key_pair.ssh-key: Creation complete after 0s [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Creation complete after 2s [id=vpc-0e839e1cef288a6f9]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creating...
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creating...
module.myapp-webserver.aws_security_group.web_sg: Creating...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creation complete after 0s [id=igw-0a6972dfa758
a7d18]
module.myapp-subnet.aws_default_route_table.main_rt: Creating...
module.myapp-subnet.aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-000ffa40f37
4bb06b]
module.myapp-webserver.aws_security_group.web_sg: Creation complete after 3s [id=sg-0429ca27d03d44a
61]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Still creating... [00m10s elapsed]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creation complete after 11s [id=subnet-0a1def6e0e1e1
7e9b]
module.myapp-webserver.aws_instance.myapp-server: Creating...
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Creation complete after 12s [id=i-0d88a073f384c72
93]

Apply complete! Resources: 7 added, 0 changed, 0 destroyed.

Outputs:

webserver_public_ip = "3.29.50.116"
@tehreem-0514 ~/Lab12 $
```

```
@tehreem-0514 ~/Lab12 $ terraform output
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ~/Lab12 $
```





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.

Task 7:

```
@tehreem-0514 ~/Lab12 $ nano apache.sh
@tehreem-0514 ~/Lab12 $ @tehreem-0514 ~/Lab12 $ chmod +x apache.sh
@tehreem-0514 ~/Lab12 $ cat apache.sh
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd

echo "<h1>Welcome to My Web Server</h1>" > /var/www/html/index.html

hostnamectl set-hostname myapp-webserver
echo "<h2>Hostname: $(hostname)</h2>" >> /var/www/html/index.html

TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" \
  -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")

echo "<h2>Private IP: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest
/meta-data/local-ipv4)</h2>" >> /var/www/html/index.html
echo "<h2>Public IP: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/public-ipv4)</h2>" >> /var/www/html/index.html
echo "<h2>Public DNS: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest
/meta-data/public-hostname)</h2>" >> /var/www/html/index.html
echo "<h2>Deployed via Terraform</h2>" >> /var/www/html/index.html

@tehreem-0514 ~/Lab12 $
```

```
 GNU nano 7.2                                  main.tf *
}

resource "aws_vpc" "myapp_vpc" {
  cidr_block = var.vpc_cidr_block
  tags = {
     Name = "${var.env_prefix}-vpc"
  }
}

module "myapp-subnet" {
  source = "./modules/subnet"

  vpc_id                  = aws_vpc.myapp_vpc.id
  subnet_cidr_block       = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  env_prefix              = var.env_prefix
  default_route_table_id  = aws_vpc.myapp_vpc.default_route_table_id
}

module "myapp-webserver" {
  source            = "./modules/webserver"
  env_prefix        = var.env_prefix
  instance_type     = var.instance_type
  availability_zone = var.availability_zone
  public_key        = var.public_key
  my_ip             = local.my_ip
  vpc_id            = aws_vpc.myapp_vpc.id
  subnet_id         = module.myapp-subnet.subnet.id
  script_path       = "./entry-script.sh"
  instance_suffix   = "0"
}

module "myapp-web-1" {
  source            = "./modules/webserver"
  env_prefix        = var.env_prefix
  instance_type     = var.instance_type
  availability_zone = var.availability_zone
  public_key        = var.public_key
  my_ip             = local.my_ip
  vpc_id            = aws_vpc.myapp_vpc.id
  subnet_id         = module.myapp-subnet.subnet.id
  script_path       = "./apache.sh"
  instance_suffix   = "1"
}
```

```
  GNU nano 7.2                                        outputs.tf *
output "webserver_public_ip" {
  value = module.myapp-webserver.aws_instance.public_ip
}

output "aws_web-1_public_ip" {
  value = module.myapp-web-1.aws_instance.public_ip
}
```

```
      + tags_all                = {
          + "Name" = "dev-web-sg-1"
        }
      + vpc_id                  = "vpc-0e839e1cef288a6f9"
    }

Plan: 3 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_web-1_public_ip = (known after apply)
module.myapp-web-1.aws_key_pair.ssh-key: Creating...
module.myapp-web-1.aws_security_group.web_sg: Creating...
module.myapp-web-1.aws_key_pair.ssh-key: Creation complete after 0s [id=dev-serverkey-1]
module.myapp-web-1.aws_security_group.web_sg: Creation complete after 3s [id=sg-030b3c01f0c4db133]
module.myapp-web-1.aws_instance.myapp-server: Creating...
module.myapp-web-1.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-web-1.aws_instance.myapp-server: Creation complete after 12s [id=i-022552bfc8c0dc231]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

Outputs:

aws_web-1_public_ip = "158.252.77.232"
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ~/Lab12 $
```

```
@tehreem-0514 ~/Lab12 $ terraform output
aws_web-1_public_ip = "158.252.77.232"
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ~/Lab12 $
```

```
@tehreem-0514 ~/Lab12 $ terraform output
aws_web-1_public_ip = "158.252.77.232"
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ~/Lab12 $ ssh ec2-user
ssh: Could not resolve hostname ec2-user: Name or service not known
@tehreem-0514 ~/Lab12 $ ssh ec2-user@3.29.50.116
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sat Jan  3 20:12:58 2026 from 4.240.18.226
[ec2-user@ip-10-0-10-76 ~]$
```

```
  GNU nano 8.3                        /etc/nginx/nginx.conf                 Modified
}

http {
    log_format  main  '$remote_addr - $remote_user [$time_local] "$request"'
                      '$status $body_bytes_sent "$http_referer"'
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile            on;
    tcp_nopush          on;
    keepalive_timeout   65;
    types_hash_max_size 4096;

    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    upstream backend_servers {
        server 158.252.94.241:80;
        server 158.252.94.242:80 backup;
    }

    server {
        listen 443 ssl;
        server_name 3.29.50.116;
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

                location / {
    #       root /usr/share/nginx/html;
    #       index index. html;
            proxy_pass http://<web-1-public-ip>:80;
    #      proxy_pass http://backend_servers;

        }

    server {
        listen 80;
        server_name _;
        return 301 https://$host$request_uri;
    }
}
```

```
[ec2-user@ip-10-0-10-76 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-76 ~]$
```

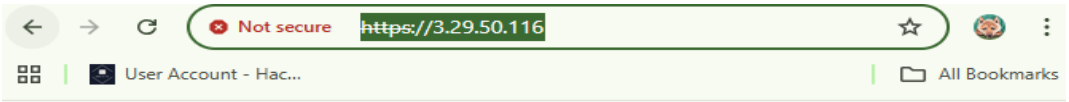
```
2026/01/03 20:13:29 [notice] 25598#25598: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2026/01/03 20:13:29 [notice] 25598#25598: getrlimit(RLIMIT_NOFILE): 65535:65535
2026/01/03 20:13:29 [notice] 25599#25599: start worker processes
2026/01/03 20:13:29 [notice] 25599#25599: start worker process 25600
2026/01/03 20:13:29 [notice] 25599#25599: start worker process 25601
2026/01/03 20:14:13 [error] 25600#25600: *5 open() "/usr/share/nginx/html/favicon.ico" failed (2: N
o such file or directory), client: 182.184.193.29, server: 3.29.50.116, request: "GET /favicon.ico
HTTP/1.1", host: "3.29.50.116", referrer: "https://3.29.50.116/"
2026/01/03 20:29:58 [notice] 25599#25599: signal 3 (SIGQUIT) received from 1, shutting down
2026/01/03 20:29:58 [notice] 25600#25600: gracefully shutting down
2026/01/03 20:29:58 [notice] 25601#25601: gracefully shutting down
2026/01/03 20:29:58 [notice] 25600#25600: exiting
2026/01/03 20:29:58 [notice] 25601#25601: exiting
2026/01/03 20:29:58 [notice] 25600#25600: exit
2026/01/03 20:29:58 [notice] 25601#25601: exit
2026/01/03 20:29:58 [notice] 25599#25599: signal 17 (SIGCHLD) received from 25601
2026/01/03 20:29:58 [notice] 25599#25599: worker process 25600 exited with code 0
2026/01/03 20:29:58 [notice] 25599#25599: worker process 25601 exited with code 0
2026/01/03 20:29:58 [notice] 25599#25599: exit
2026/01/03 20:29:58 [emerg] 26197#26197: host not found in upstream "<web-1-public-ip>" in /etc/ngi
nx/nginx.conf:39
2026/01/03 20:30:17 [emerg] 26205#26205: host not found in upstream "<web-1-public-ip>" in /etc/ngi
nx/nginx.conf:39
2026/01/03 20:33:13 [notice] 26319#26319: using the "epoll" event method
2026/01/03 20:33:13 [notice] 26319#26319: nginx/1.28.0
2026/01/03 20:33:13 [notice] 26319#26319: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2026/01/03 20:33:13 [notice] 26319#26319: getrlimit(RLIMIT_NOFILE): 65535:65535
2026/01/03 20:33:13 [notice] 26320#26320: start worker processes
2026/01/03 20:33:13 [notice] 26320#26320: start worker process 26321
2026/01/03 20:33:13 [notice] 26320#26320: start worker process 26322
2026/01/03 20:33:59 [notice] 26320#26320: signal 3 (SIGQUIT) received from 1, shutting down
2026/01/03 20:33:59 [notice] 26322#26322: gracefully shutting down
2026/01/03 20:33:59 [notice] 26321#26321: gracefully shutting down
2026/01/03 20:33:59 [notice] 26322#26322: exiting
2026/01/03 20:33:59 [notice] 26321#26321: exiting
2026/01/03 20:33:59 [notice] 26322#26322: exit
2026/01/03 20:33:59 [notice] 26321#26321: exit
2026/01/03 20:33:59 [notice] 26320#26320: signal 17 (SIGCHLD) received from 26321
2026/01/03 20:33:59 [notice] 26320#26320: worker process 26321 exited with code 0
2026/01/03 20:33:59 [notice] 26320#26320: worker process 26322 exited with code 0
2026/01/03 20:33:59 [notice] 26320#26320: exit
2026/01/03 20:33:59 [notice] 26382#26382: using the "epoll" event method
2026/01/03 20:33:59 [notice] 26382#26382: nginx/1.28.0
2026/01/03 20:33:59 [notice] 26382#26382: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2026/01/03 20:33:59 [notice] 26382#26382: getrlimit(RLIMIT_NOFILE): 65535:65535
2026/01/03 20:33:59 [notice] 26384#26384: start worker processes
2026/01/03 20:33:59 [notice] 26384#26384: start worker process 26385
2026/01/03 20:33:59 [notice] 26384#26384: start worker process 26386
[ec2-user@ip-10-0-10-76 ~]$
```

```
[ec2-user@ip-10-0-10-76 ~]$ cat /var/log/nginx/access.log
182.184.193.29 - - [03/Jan/2026:20:14:13 +0000] "GET / HTTP/1.1"200 615 "-""Mozilla/5.0 (Windows NT
 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-"
182.184.193.29 - - [03/Jan/2026:20:14:13 +0000] "GET /favicon.ico HTTP/1.1"404 555 "https://3.29.50
.116/""Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.
0.0.0 Safari/537.36" "-"
185.16.39.146 - - [03/Jan/2026:20:20:12 +0000] "GET / HTTP/1.1"301 169 "-""Wget" "-"
185.16.39.146 - - [03/Jan/2026:20:28:28 +0000] "GET / HTTP/1.1"301 169 "-""Wget" "-"
[ec2-user@ip-10-0-10-76 ~]$
```

```
application/x-xpinstall                         xpi;
application/x-xspf+xml                          xspf;
application/x-xz                                xz;
audio/midi                                      mid midi kar;
audio/x-aiff                                    aif aiff aifc;
audio/x-annodex                                 axa;
audio/x-flac                                    flac;
audio/x-matroska                                mka;
audio/x-mod                                     mod ult uni m15 mtm 669 med;
audio/x-mpegurl                                 m3u;
audio/x-ms-wax                                  wax;
audio/x-ms-wma                                  wma;
audio/x-pn-realaudio                            ram rm;
audio/x-realaudio                               ra;
audio/x-s3m                                     s3m;
audio/x-stm                                     stm;
audio/x-wav                                     wav;
chemical/x-xyz                                  xyz;
image/webp                                      webp;
image/x-cmu-raster                              ras;
image/x-portable-anymap                         pnm;
image/x-portable-bitmap                         pbm;
image/x-portable-graymap                        pgm;
image/x-portable-pixmap                         ppm;
image/x-rgb                                     rgb;
image/x-targa                                   tga;
image/x-xbitmap                                 xbm;
image/x-xpixmap                                 xpm;
image/x-xwindowdump                             xwd;
text/html-sandboxed                             sandboxed;
text/x-pod                                      pod;
text/x-setext                                   etx;
video/webm                                      webm;
video/x-annodex                                 axv;
video/x-flv                                     flv;
video/x-javafx                                  fxm;
video/x-matroska                                mkv;
video/x-matroska-3d                             mk3d;
video/x-ms-asf                                  asx;
video/x-ms-wm                                   wm;
video/x-ms-wmv                                  wmv;
video/x-ms-wmx                                  wmx;
video/x-ms-wvx                                  wvx;
video/x-msvideo                                 avi;
video/x-sgi-movie                               movie;
x-conference/x-cooltalk                         ice;
x-epoc/x-sisx-app                               sisx;
}
[ec2-user@ip-10-0-10-76 ~]$
```

```
[ec2-user@ip-10-0-10-76 ~]$ [ec2-user@ip-10-0-10-76 ~]$ cat /etc/ssl/certs/selfsigned.crt
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIUKBjATueVhx+lCUHpNNbEFHIzMq4wDQYJKoZIhvcNAQEL
BQAwFjEUMBIGA1UEAwwLMy4yOS41MC4xMTYwHhcNMjYwMTAzMjAwNDM0WhcNMjcw
MTAzMjAwNDM0WjAWMRQwEgYDVQQDDAszLjI5LjUwLjExNjCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBAJimVWcOK2j4QiDsLcooJ2sS1QirT1vQ7NIm+Uyj
ApoYlAJrcWVKn8C+H3C8DLzXEol8YJzVNByiAZ8ZNVpb7SwY8yIXwUThwNFIuWuf
r9SMoMk/veGqxwVFIPVg3eVMGt7q0tPEfkkYf16sEPFYd1wcCuThA0daA0TKEgD9
LtVWbj3jU64FzgkpbdKoFIrmA8TVvMr+dB06f8yqFTMaR8bsDBLx87WUyyxHYJ49
SoLpLXibgBADSdvZtvWXxM0QXosIH/1ADebTMllC17aJb98r2nvTDIx7lndXpm/p
Og1QJqIhGHyv7UUN671WM9NKO7RXH8irT8dzs1OOLpgkMnMCAwEAAaOBgDB+MB0G
A1UdDgQWBBQoSmQaNTU10L+017puycBQ4+xExTAfBgNVHSMEGDAWgBQoSmQaNTU1
0L+017puycBQ4+xExTAPBgNVHREECDAGhwQDHTJ0MAkGA1UdEwQCMAAwCwYDVR0P
BAQDAgWgMBMGA1UdJQQMMAoGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUAA4IBAQAT
/hdbo/F5FAshqnUJjOsHAWjf/s19VcSo/fE3QREzEtbH1mnC1jzM6dpv9zJzreRL
WN7fymLu4z3mc7h/yPzV1d8h9b5/7b9+jvzNxshbAhHu8bdEmQWZ9UmkP6gK2LCF
vZrV/s1w79R8vxd7Qc3PlkUzOwLpm3JhVArTTP1con3hE94+USgJwi1V8uXjQcZn
Hjy+cEMab2AiEPfFtyTQkIfm0HxOWtouDNay5Yea+TH0XEucjfrwzkNdzBFyMSZ+
4TMN8SbHSeC9rBe+b2NXUoKAWZvEChcP6bt1Pwe6P3JZVBx/DSuGjdyhYUIC3kyE
1Arybwg5ckE2pAhVk9Vv
-----END CERTIFICATE-----
[ec2-user@ip-10-0-10-76 ~]$
```

```
[ec2-user@ip-10-0-10-76 ~]$ sudo cat /etc/ssl/private/selfsigned.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBK
7C3KKC
GTVaW+0sGPMiF8FE4cDR
rBD
qhUz
wte
ji6
5ag
Es
nLEI
kPxWx
CUtqI
iDsjyc
T6ohUwC
a+xv8D
Oy0C
sN1a
1dmnuM
6pVPV+e+HsCv
NrWwB47W5E8H
hLWWqntUgGWw
QNzNsNPVzAV
Q9Ug4x
5tfO1EGNkQxKR3GG5GKeLuWenk02K2FDCrf
f+JesJg
3rbRXTWoSoRWc/MGLRkCRLE=
-----END PRIVATE KEY-----
[ec2-user@ip-10-0-10-76 ~]$
```



Welcome to My Web Server

Hostname: myapp-webserver

Private IP: 10.0.10.32

Public IP: 158.252.77.232

Public DNS:

Deployed via Terraform

Task 8:

```
  GNU nano 7.2                                  main.tf
  subnet_cidr_block       = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  env_prefix              = var.env_prefix
  default_route_table_id  = aws_vpc.myapp_vpc.default_route_table_id
}

module "myapp-webserver" {
  source             = "./modules/webserver"
  env_prefix         = var.env_prefix
  instance_type      = var.instance_type
  availability_zone  = var.availability_zone
  public_key         = var.public_key
  my_ip              = local.my_ip
  vpc_id             = aws_vpc.myapp_vpc.id
  subnet_id          = module.myapp-subnet.subnet.id
  script_path        = "./entry-script.sh"
  instance_suffix    = "0"
}

module "myapp-web-1" {
  source             = "./modules/webserver"
  env_prefix         = var.env_prefix
  instance_type      = var.instance_type
  availability_zone  = var.availability_zone
  public_key         = var.public_key
  my_ip              = local.my_ip
  vpc_id             = aws_vpc.myapp_vpc.id
  subnet_id          = module.myapp-subnet.subnet.id
  script_path        = "./apache.sh"
  instance_suffix    = "1"
}

module "myapp-web-2" {
  source = "./modules/webserver"
  env_prefix = var.env_prefix
  instance_type = var.instance_type
  availability_zone = var. availability_zone
  public_key = var.public_key
  my_ip = local.my_ip
  vpc_id = aws_vpc.myapp_vpc.id
  subnet_id = module.myapp-subnet.subnet.id
  script_path = "./apache.sh"
  instance_suffix = "2"
}
```

```
  GNU nano 7.2                                  outputs.tf *
output "webserver_public_ip" {
  value = module.myapp-webserver.aws_instance.public_ip
}

output "aws_web-1_public_ip" {
  value = module.myapp-web-1.aws_instance.public_ip
}

output "aws_web-2_public_ip" {
  value = module. myapp-web-2.aws_instance.public_ip
}
```

```
Plan: 3 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + aws_web-2_public_ip = (known after apply)
module.myapp-web-2.aws_key_pair.ssh-key: Creating...
module.myapp-web-2.aws_security_group.web_sg: Creating...
module.myapp-web-2.aws_key_pair.ssh-key: Creation complete after 1s [id=dev-serverkey-2]
module.myapp-web-2.aws_security_group.web_sg: Creation complete after 3s [id=sg-042c4b9aa6e2f3
61f]
module.myapp-web-2.aws_instance.myapp-server: Creating...
module.myapp-web-2.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-web-2.aws_instance.myapp-server: Creation complete after 13s [id=i-0fbc63e1106b98
a42]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.

Outputs:

aws_web-1_public_ip = "158.252.77.232"
aws_web-2_public_ip = "3.28.184.78"
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ▨ ~/Lab12 $
```

```
@tehreem-0514 ▨ ~/Lab12 $ terraform output
aws_web-1_public_ip = "158.252.77.232"
aws_web-2_public_ip = "3.28.184.78"
webserver_public_ip = "3.29.50.116"
@tehreem-0514 ▨ ~/Lab12 $
```

```
pid /run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    log_format  main  '$remote_addr - $remote_user [$time_local] "$request"'
                      '$status $body_bytes_sent "$http_referer"'
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile            on;
    tcp_nopush          on;
    keepalive_timeout   65;
    types_hash_max_size 4096;

    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    upstream backend_servers {
        server 158.252.77.232:80;
        server 3.28.184.78:80 backup;
    }

    server {
        listen 443 ssl;
        server_name 3.29.50.116;
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

           location / {
              proxy_pass http://backend_servers;

        }
    }

    server {
        listen 80;
        server_name _;
        return 301 https://$host$request_uri;
    }
}

^G Help        ^O Write Out   ^F Where Is    ^K Cut         ^T Execute     ^C Location
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line
```

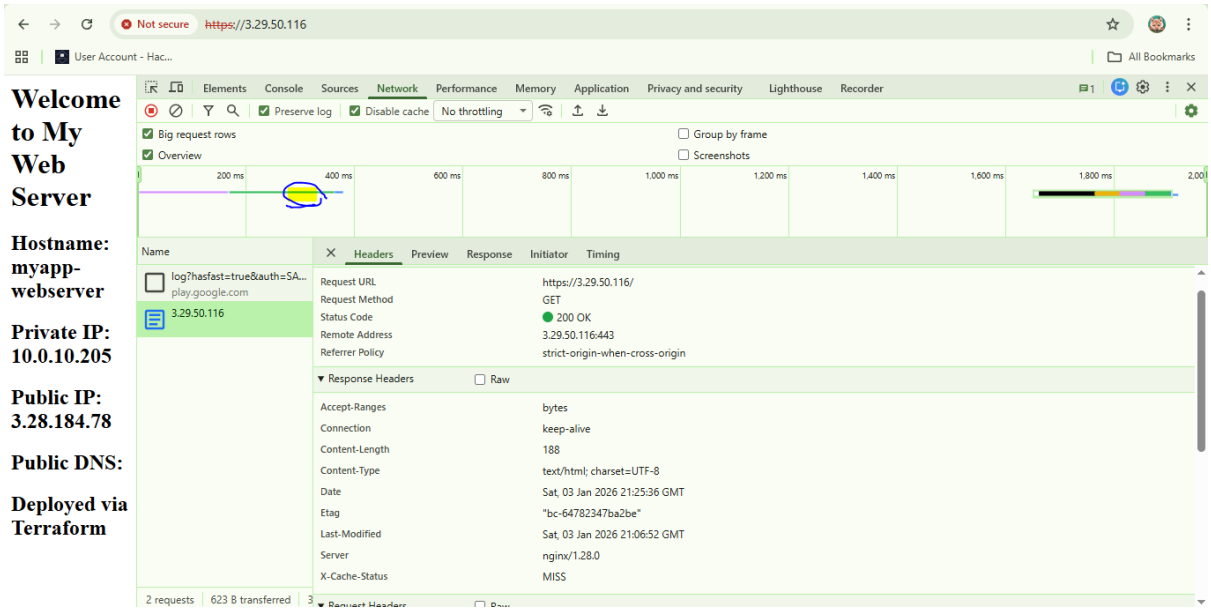

Not secure
https://3.29.50.116
User Account - Hac...
Welcome to My Web Server
Hostname: myapp-webserver
Private IP: 10.0.10.32
Public IP: 158.252.77.232
Public DNS:
Deployed via Terraform

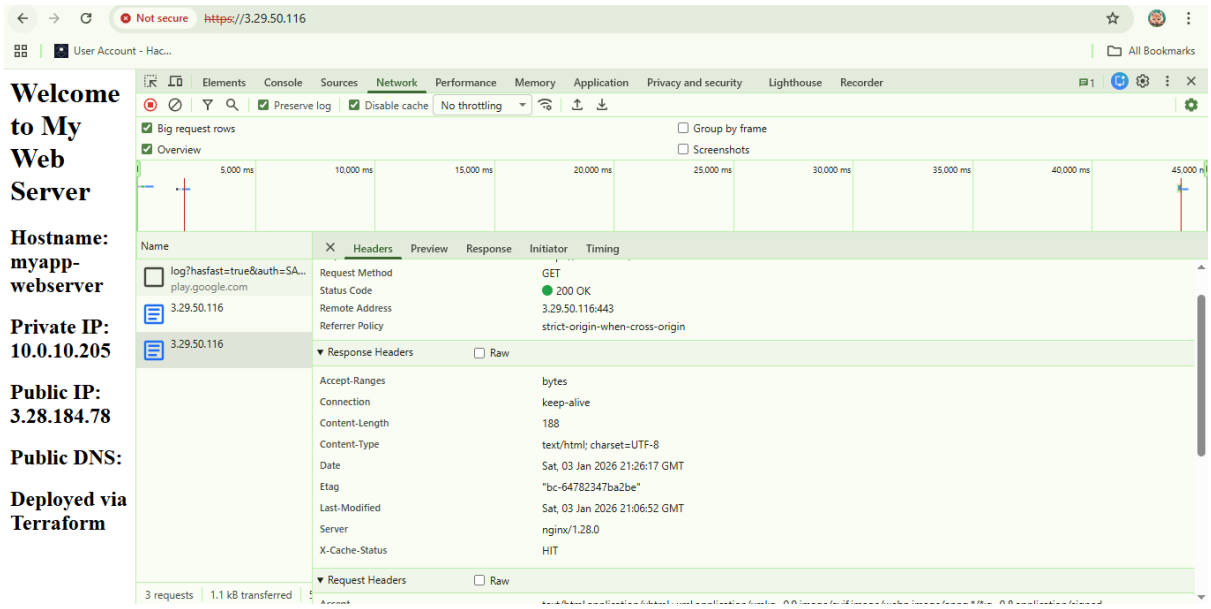

Not secure
https://3.29.50.116
User Account - Hac...
Welcome to My Web Server
Hostname: myapp-webserver
Private IP: 10.0.10.205
Public IP: 3.28.184.78
Public DNS:
Deployed via Terraform

Task 10:

```
    include             /etc/nginx/mime.types;
    default_type        application/octet-stream;

    upstream backend_servers {
        server 158.252.77.232:80 backup;
        server 3.28.184.78:80;
    }

    server {
        listen 443 ssl;
        server_name 3.29.50.116;
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

        location / {
            proxy_pass http://backend_servers;
            proxy_cache my_cache;
            proxy_cache_valid 200 60m;
            proxy_cache_key "$scheme$request_uri";
            add_header X-Cache-Status $upstream_cache_status;
        }
      }

    server {
        listen 80;
        server_name _;
        return 301 https://$host$request_uri;
```

```
[ec2-user@ip-10-0-10-76 ~]$ [ec2-user@ip-10-0-10-76 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-76 ~]$
```






```
[ec2-user@ip-10-0-10-76 ~]$ sudo ls -la /var/cache/nginx/
total 0
drwx------. 3 nginx root   15 Jan  3 21:25 .
drwxr-xr-x. 9 root  root  101 Jan  3 21:21 ..
drwx------. 3 nginx nginx  16 Jan  3 21:25 4
[ec2-user@ip-10-0-10-76 ~]$
```

Cleanup:

```
@tehreem-0514 /workspaces/CC_TehreemKhan_064_Lab12 (main) $ terraform destroy
```

```
Destroy complete! Resources: 0 destroyed.
@tehreem-0514 /workspaces/CC_TehreemKhan_064_Lab12 (main) $
```

```
@tehreem-0514 /workspaces/CC_TehreemKhan_064_Lab12 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 1,
  "lineage": "6ea6e977-d8aa-74f6-8ea9-319b4abee520",
  "outputs": {},
  "resources": [],
  "check_results": null
}
```

```
@tehreem-0514 /workspaces/CC_TehreemKhan_064_Lab12 (main) $ tree
.
├── README.md
└── terraform.tfstate

1 directory, 2 files
@tehreem-0514 /workspaces/CC_TehreemKhan_064_Lab12 (main) $
```