CLOUD COMPUTING LAB

Fatima Jinnah
Women University

SUBMITTED TO
SIR WAQAS SALEEM & SIR SHOAIB

SUBMITTED BY
HAMNA MAHMOOD
2023-BSE-025

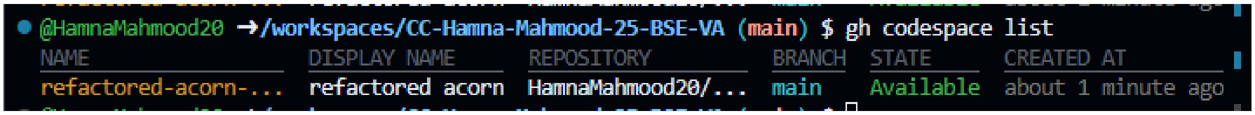
BSE V-A

Lab 12

Terraform Provisioners, Modules & Nginx Reverse Proxy/Load Balancer

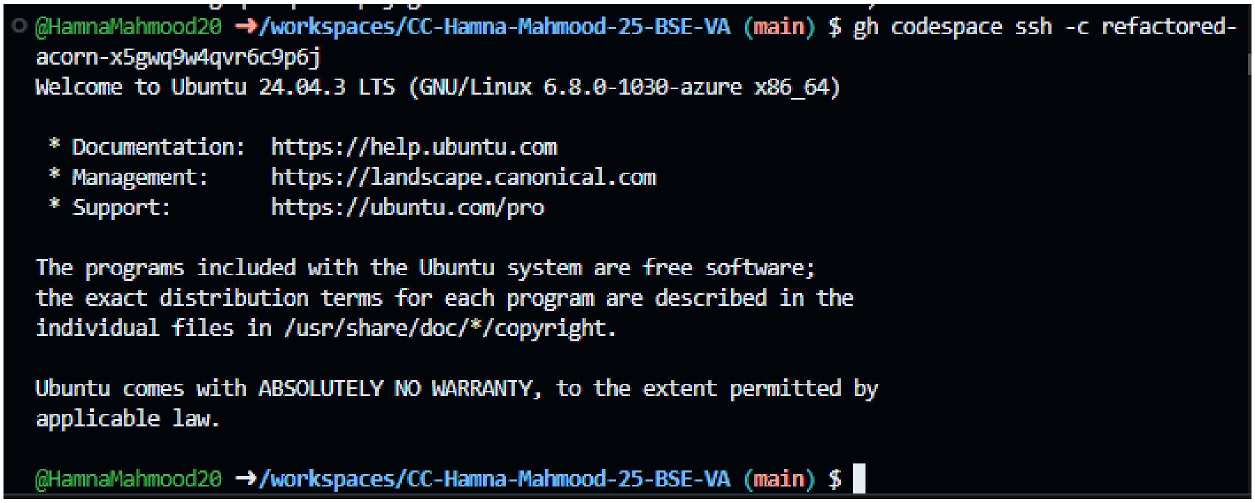
Task 0 Lab Setup (Codespace & GH CLI)

- task0_codespace_create_and_list. Png



```
● @HamnaMahmood20 ➜/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ gh codespace list
  NAME                  DISPLAY NAME       REPOSITORY          BRANCH  STATE      CREATED AT
  refactored-acorn-...  refactored acorn  HamnaMahmood20/...  main    Available  about 1 minute ago
```

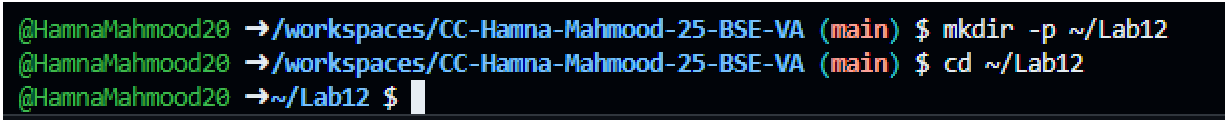
- task0_codespace_ssh_connected.png



```
○ @HamnaMahmood20 ➜/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ gh codespace ssh -c refactored-
  acorn-x5gwq9w4qvr6c9p6j
  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.

  @HamnaMahmood20 ➜/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
```

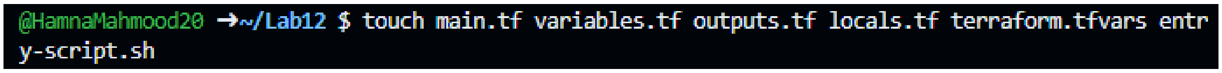
Task 1 – Organize Terraform code into separate files

- task1_project_directory.png

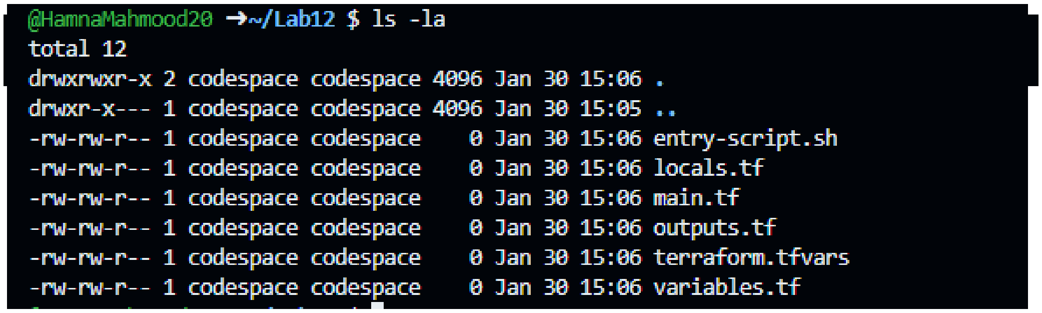


```
@HamnaMahmood20 ➜/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ mkdir -p ~/Lab12
@HamnaMahmood20 ➜/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cd ~/Lab12
@HamnaMahmood20 ➜~/Lab12 $
```

- task1_files_created.png

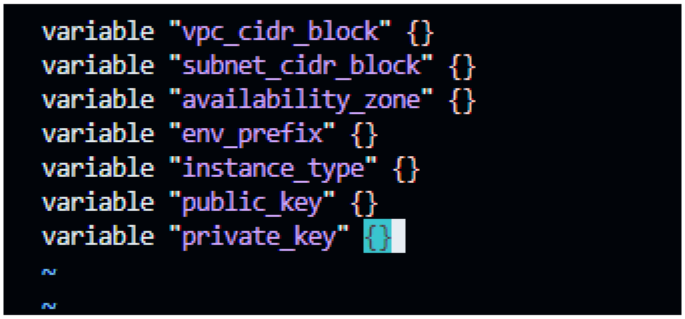


```
@HamnaMahmood20 ➜~/Lab12 $ touch main.tf variables.tf outputs.tf locals.tf terraform.tfvars entr
y-script.sh
```



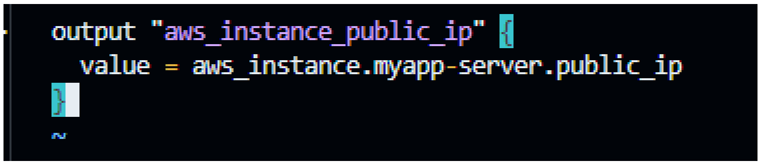
```
@HamnaMahmood20 ➜~/Lab12 $ ls -la
total 12
drwxrwxr-x 2 codespace codespace 4096 Jan 30 15:06 .
drwxr-x--- 1 codespace codespace 4096 Jan 30 15:05 ..
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 entry-script.sh
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 locals.tf
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 main.tf
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 outputs.tf
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 terraform.tfvars
-rw-rw-r-- 1 codespace codespace    0 Jan 30 15:06 variables.tf
```

- task1_variables_tf.png



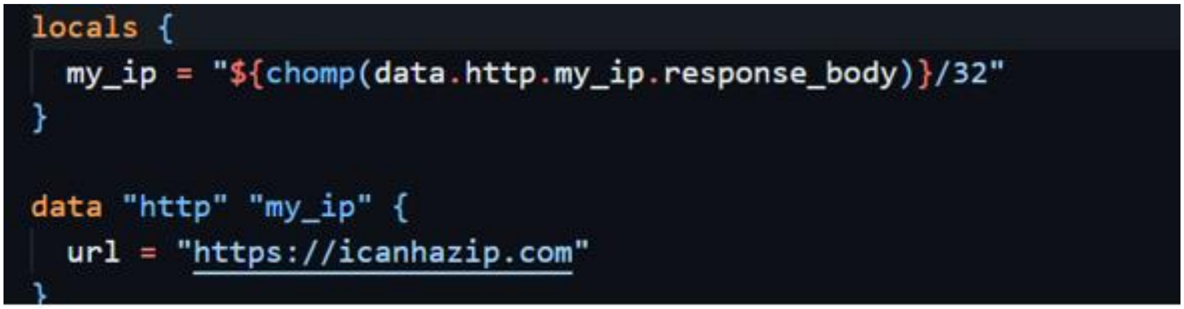
```
variable "vpc_cidr_block" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
variable "instance_type" {}
variable "public_key" {}
variable "private_key" {}
~
~
```

- task1_outputs_tf.png



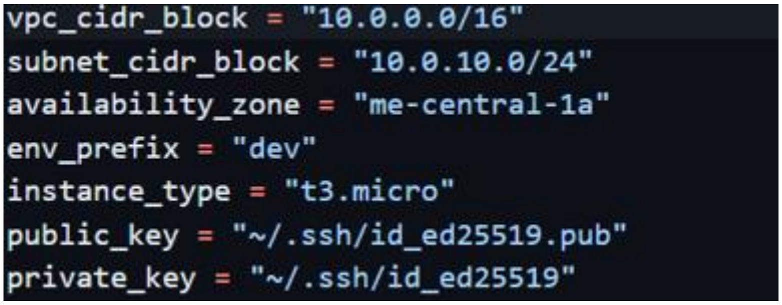
```
output "aws_instance_public_ip" {
  value = aws_instance.myapp-server.public_ip
}
~
```

- task1_locals_tf.png



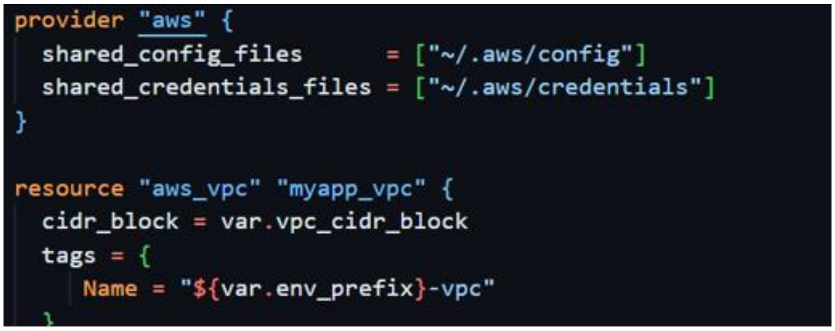
```
locals {
  my_ip = "${chomp(data.http.my_ip.response_body)}/32"
}

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

- task1_terraform_tfvars.png



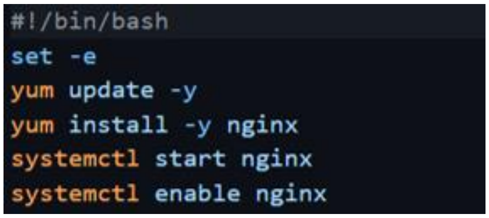
```
vpc_cidr_block = "10.0.0.0/16"
subnet_cidr_block = "10.0.10.0/24"
availability_zone = "me-central-1a"
env_prefix = "dev"
instance_type = "t3.micro"
public_key = "~/.ssh/id_ed25519.pub"
private_key = "~/.ssh/id_ed25519"
```

- task1_main_tf.png



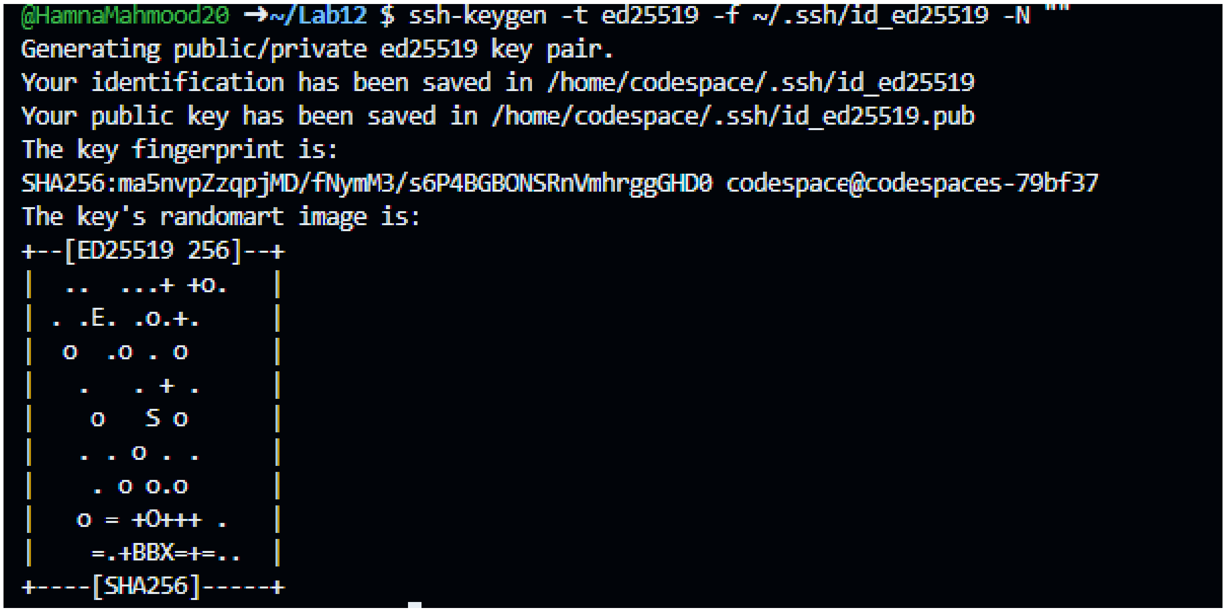
```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

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

- task1_entry_script.png



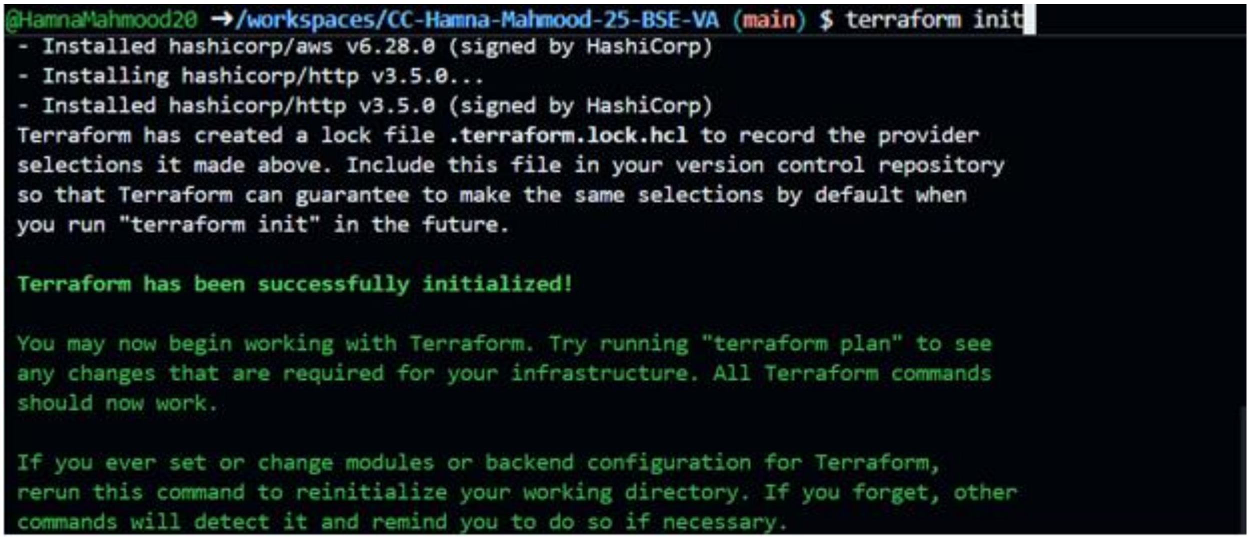
```
#!/bin/bash
set -e
yum update -y
yum install -y nginx
systemctl start nginx
systemctl enable nginx
```

- task1_ssh_keygen.png



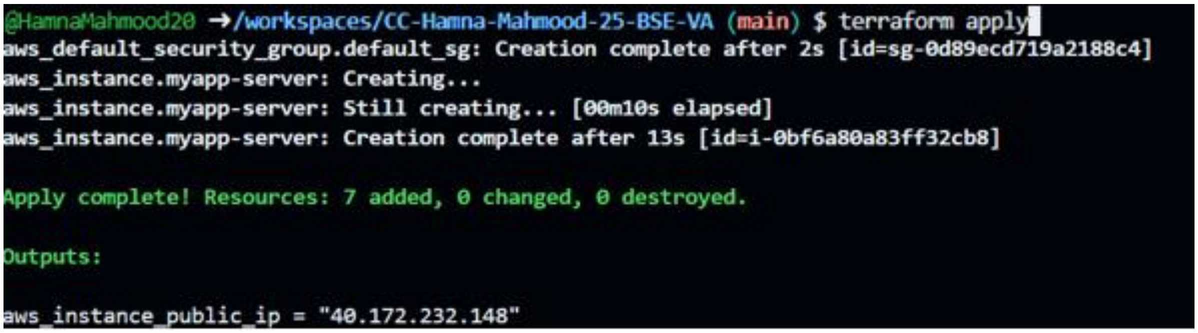
```
@HamnaMahmood20 ➜~/Lab12 $ ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -N ""
Generating public/private ed25519 key pair.
Your identification has been saved in /home/codespace/.ssh/id_ed25519
Your public key has been saved in /home/codespace/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:ma5nvpZzqpjMD/fNymM3/s6P4BGBONSRnVmhrggGHD0 codespace@codespaces-79bf37
The key's randomart image is:
+--[ED25519 256]--+
|  ..  ...+ +o.   |
| . .E. .o.+.     |
|  o  .o . o      |
|   .   . + .     |
|    o   S o      |
|   . . o . .     |
|    . o o.o      |
|   o = +O+++ .   |
|    =.+BBX=+=..  |
+----[SHA256]-----+
```

- task1_terraform_init.png



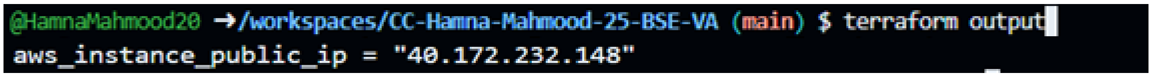
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform init
 - Installed hashicorp/aws v6.28.0 (signed by HashiCorp)
 - Installing hashicorp/http v3.5.0...
 - Installed hashicorp/http v3.5.0 (signed by HashiCorp)
 Terraform has created a lock file .terraform.lock.hcl to record 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.
```

- task1_terraform_apply.png



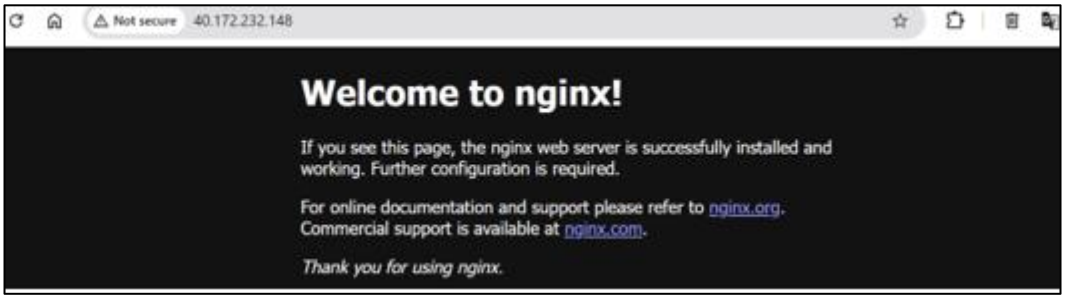
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform apply
aws_default_security_group.default_sg: Creation complete after 2s [id=sg-0d89ecd719a2188c4]
aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-0bf6a80a83ff32cb8]

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

Outputs:

aws_instance_public_ip = "40.172.232.148"
```

- task1_terraform_output.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform output
aws_instance_public_ip = "40.172.232.148"
```

- task1_nginx_browser.png



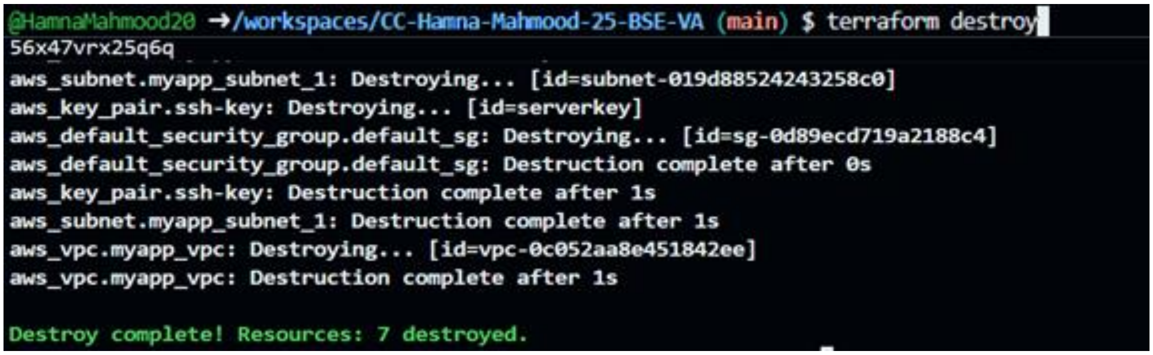
Welcome to nginx!

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.

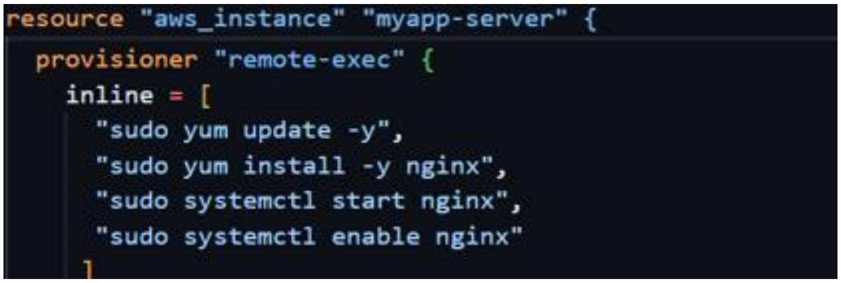
- task1_terraform_destroy.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform destroy
56x47vrx25q6q
aws_subnet.myapp_subnet_1: Destroying... [id=subnet-019d88524243258c0]
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_default_security_group.default_sg: Destroying... [id=sg-0d89ecd719a2188c4]
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-0c052aa8e451842ee]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
```

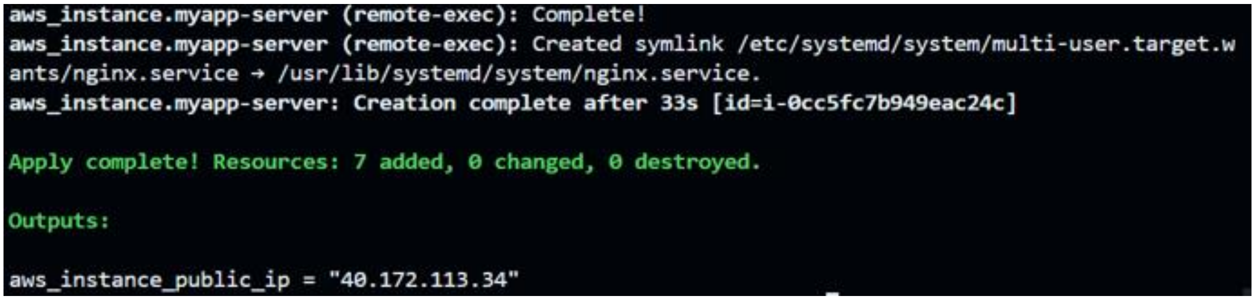
Task 2 – Use remote exec provisioner

- task2_main_tf_remote_exec.png



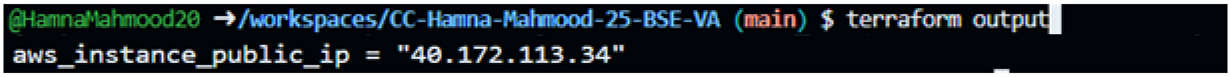
```
resource "aws_instance" "myapp-server" {
  provisioner "remote-exec" {
    inline = [
      "sudo yum update -y",
      "sudo yum install -y nginx",
      "sudo systemctl start nginx",
      "sudo systemctl enable nginx"
    ]
```

- task2_terraform_apply.png



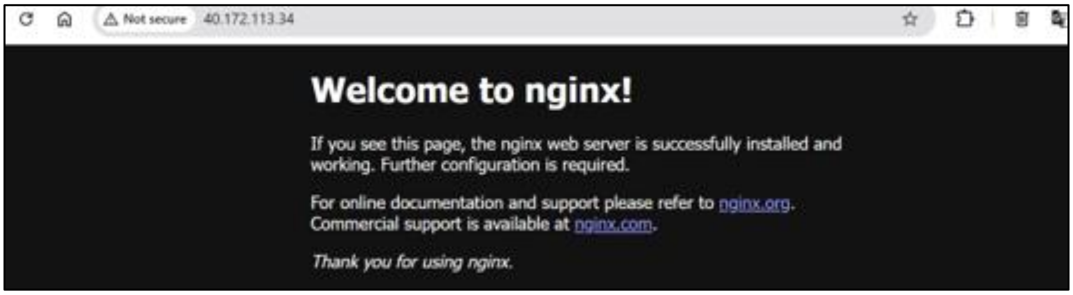
```
aws_instance.myapp-server (remote-exec): Complete!
aws_instance.myapp-server (remote-exec): Created symlink /etc/systemd/system/multi-user.target.w
ants/nginx.service → /usr/lib/systemd/system/nginx.service.
aws_instance.myapp-server: Creation complete after 33s [id=i-0cc5fc7b949eac24c]

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

Outputs:

aws_instance_public_ip = "40.172.113.34"
```

- task2_terraform_output.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform output
aws_instance_public_ip = "40.172.113.34"
```

- task2_nginx_browser.png



Welcome to nginx!

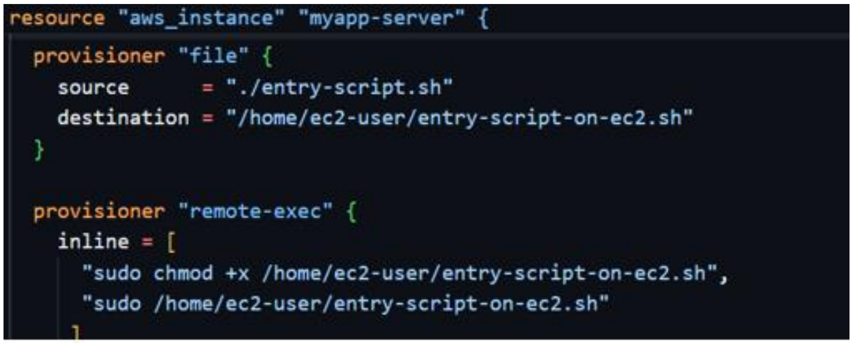
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

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

Thank you for using nginx.

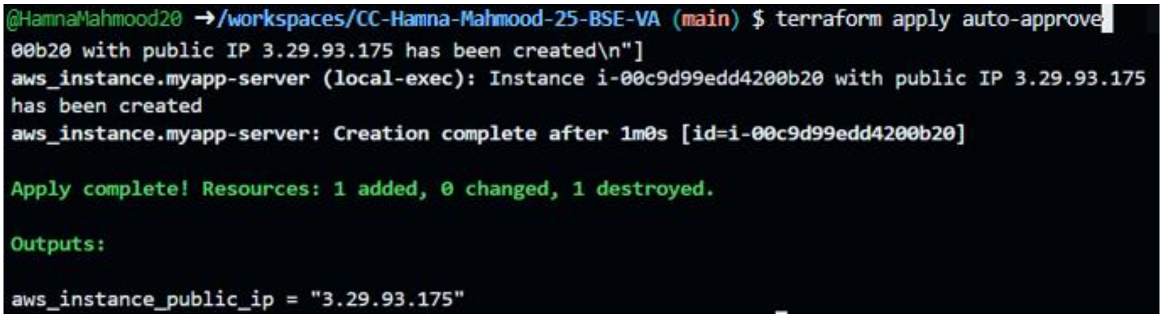
Task 3- Use file & local exec provisioner

- task3_main_tf_all_provisioners.png



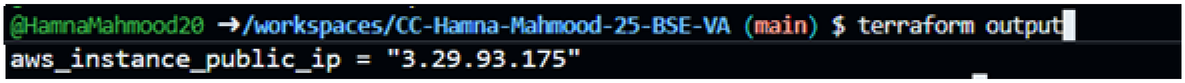
```
resource "aws_instance" "myapp-server" {
  provisioner "file" {
    source      = "./entry-script.sh"
    destination = "/home/ec2-user/entry-script-on-ec2.sh"
  }

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

- task3_terraform_apply.png



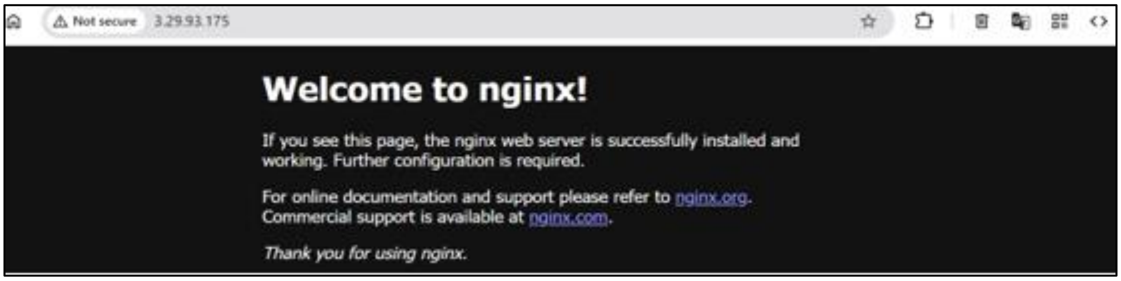
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform apply auto-approve
00b20 with public IP 3.29.93.175 has been created\n"]
aws_instance.myapp-server (local-exec): Instance i-00c9d99edd4200b20 with public IP 3.29.93.175
has been created
aws_instance.myapp-server: Creation complete after 1m0s [id=i-00c9d99edd4200b20]

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

Outputs:

aws_instance_public_ip = "3.29.93.175"
```

- task3_terraform_output.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform output
aws_instance_public_ip = "3.29.93.175"
```

- task3_nginx_browser.png



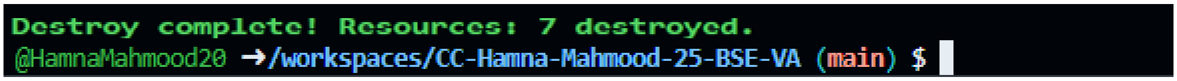
Welcome to nginx!

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.

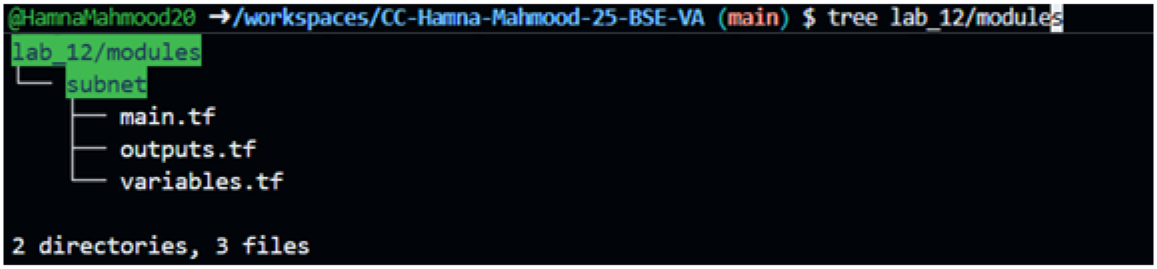
- task3_terraform_destroy.png



```
Destroy complete! Resources: 7 destroyed.
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
```

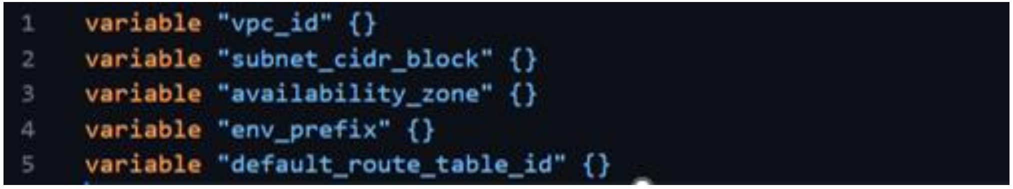
Task 4 - Create Terraform modules (subnet module)

- task4_module_structure.png



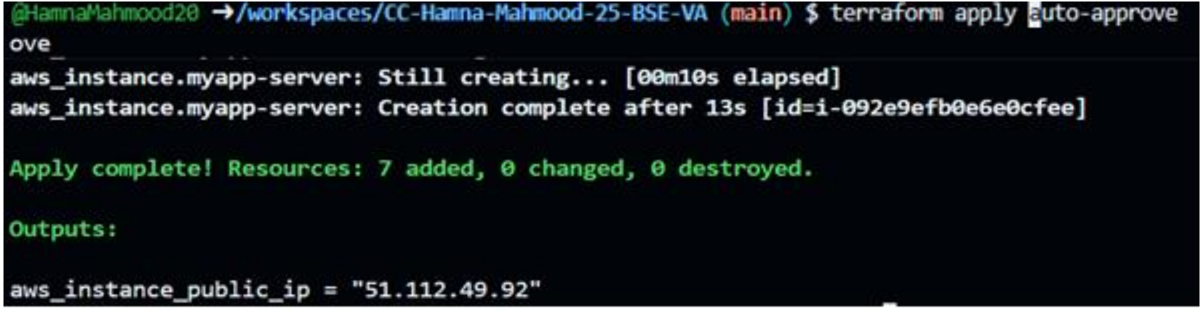
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ tree lab_12/modules
lab_12/modules
└── subnet
    ├── main.tf
    ├── outputs.tf
    └── variables.tf

2 directories, 3 files
```

- task4_subnet_variables.png



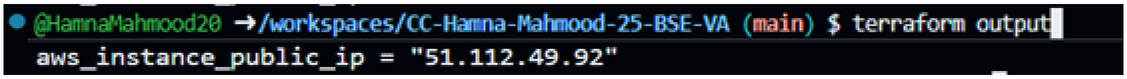
```
variable "vpc_id" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
variable "default_route_table_id" {}
```

- task4_terraform_apply.png



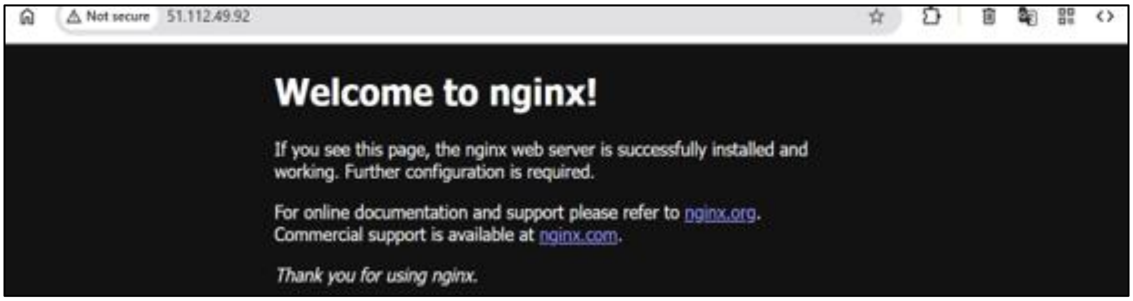
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform apply auto-approve
ove
aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Creation complete after 13s [id=i-092e9efb0e6e0cfee]

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

Outputs:

aws_instance_public_ip = "51.112.49.92"
```

- task4_terraform_output.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform output
  aws_instance_public_ip = "51.112.49.92"
```

- task4_nginx_browser.png



Welcome to nginx!

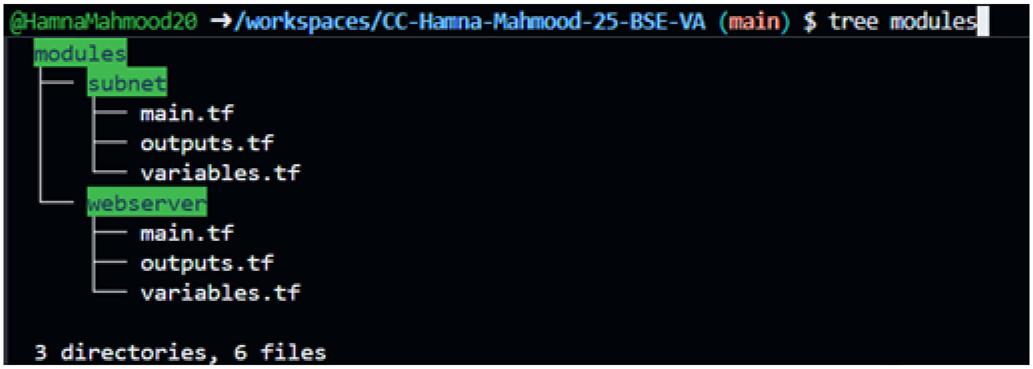
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

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

Thank you for using nginx.

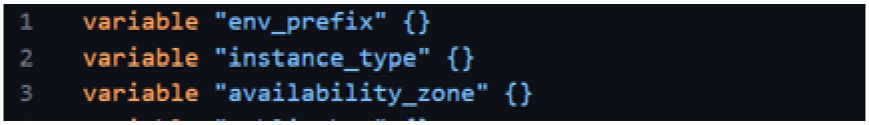
Task 5 - Create webserver module

- task5_webserver_module_structure.png



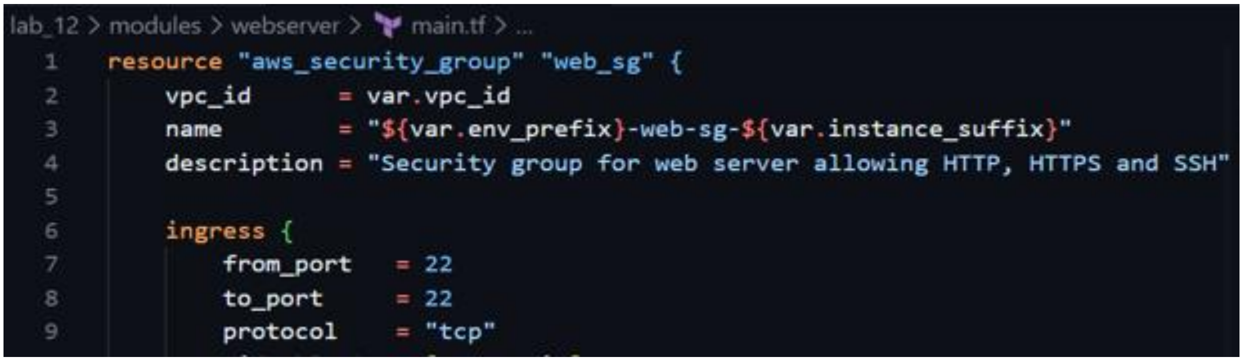
```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ tree modules
modules
├── subnet
│   ├── main.tf
│   ├── outputs.tf
│   └── variables.tf
└── webserver
    ├── main.tf
    ├── outputs.tf
    └── variables.tf

3 directories, 6 files
```

- task5_webserver_variables.png



```
variable "env_prefix" {}
variable "instance_type" {}
variable "availability_zone" {}
```

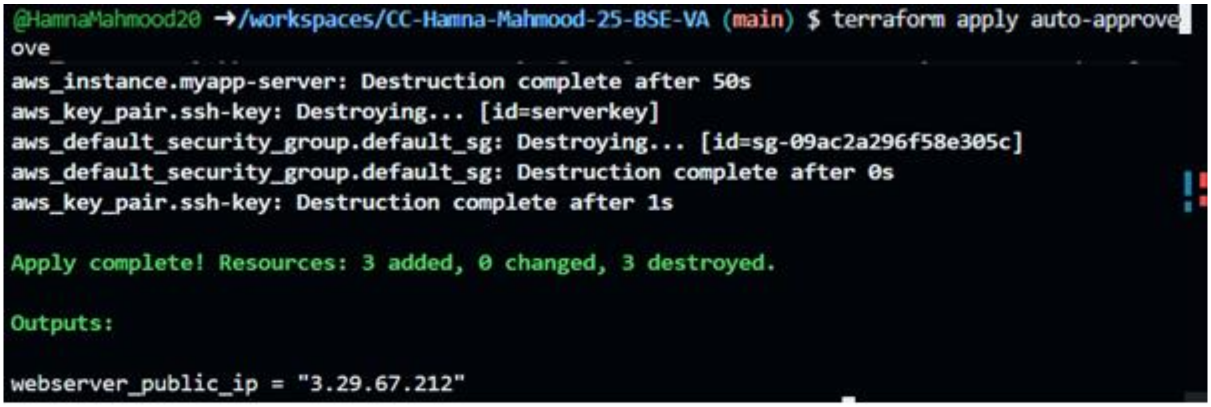
- task5_webserver_main.png



lab_12 > modules > webserver > main.tf > ...

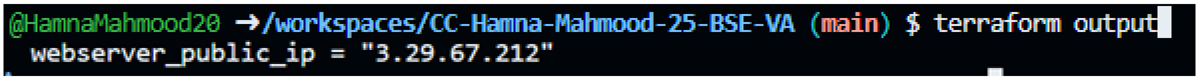
```
resource "aws_security_group" "web_sg" {
    vpc_id      = var.vpc_id
    name        = "${var.env_prefix}-web-sg-${var.instance_suffix}"
    description = "Security group for web server allowing HTTP, HTTPS and SSH"

    ingress {
        from_port    = 22
        to_port      = 22
        protocol     = "tcp"
```

- task5_terraform_apply.png



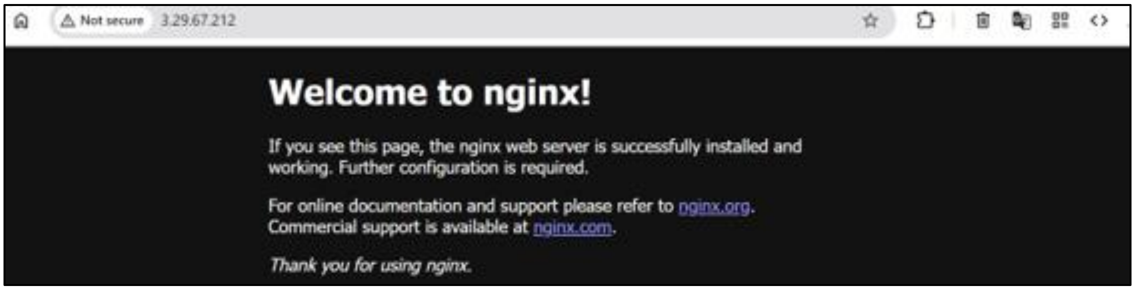
```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform apply auto-approve
ove
aws_instance.myapp-server: Destruction complete after 50s
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_default_security_group.default_sg: Destroying... [id=sg-09ac2a296f58e305c]
aws_default_security_group.default_sg: Destruction complete after 0s
aws_key_pair.ssh-key: Destruction complete after 1s

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

Outputs:

webserver_public_ip = "3.29.67.212"
```

- task5_terraform_output.png



```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform output
 webserver_public_ip = "3.29.67.212"
```

- task5_nginx_browser.png



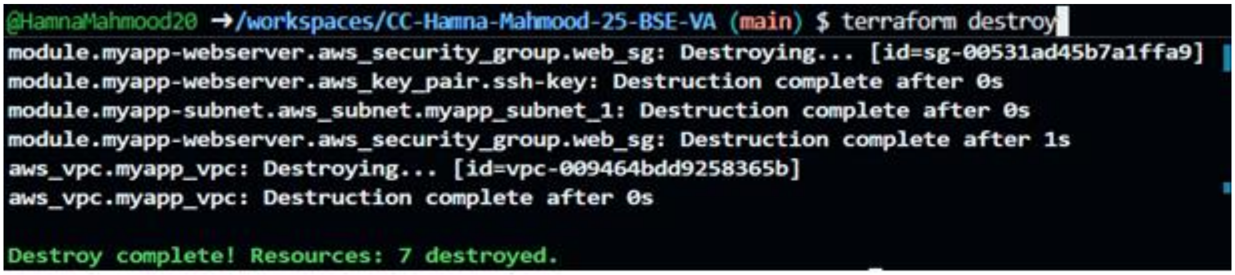
Welcome to nginx!

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.

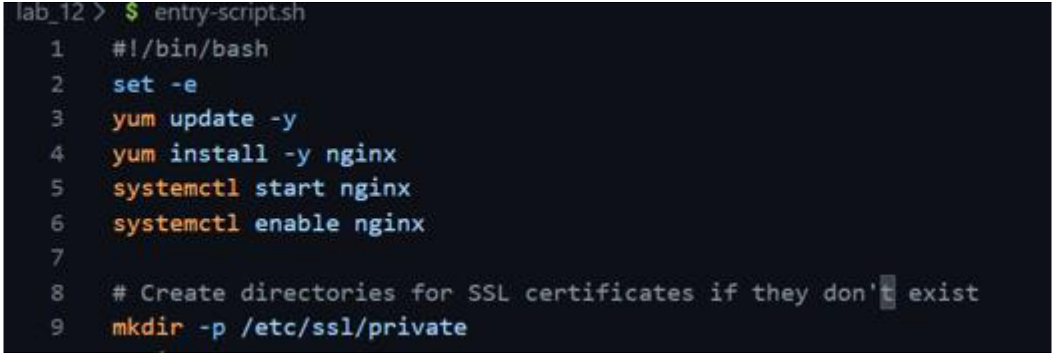
- task5_terraform_destroy.png



```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform destroy
module.myapp-webserver.aws_security_group.web_sg: Destroying... [id=sg-00531ad45b7a1ffa9]
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 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-009464bdd9258365b]
aws_vpc.myapp_vpc: Destruction complete after 0s

Destroy complete! Resources: 7 destroyed.
```

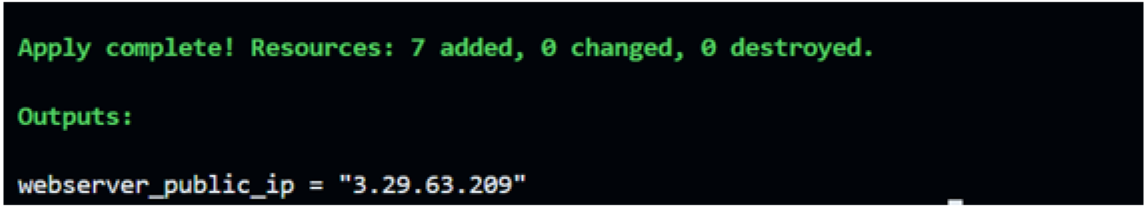
Task 6 - Configure HTTPS with self-signed certificates

- task6_entry_script_https.png



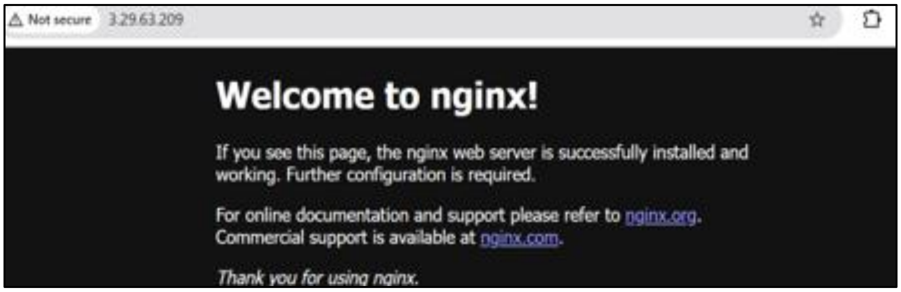
```
#!/bin/bash
set -e
yum update -y
yum install -y nginx
systemctl start nginx
systemctl enable nginx

# Create directories for SSL certificates if they don't exist
mkdir -p /etc/ssl/private
```

- task6_terraform_apply.png



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

Outputs:

webserver_public_ip = "3.29.63.209"
```

- task6_nginx_https_browser.png



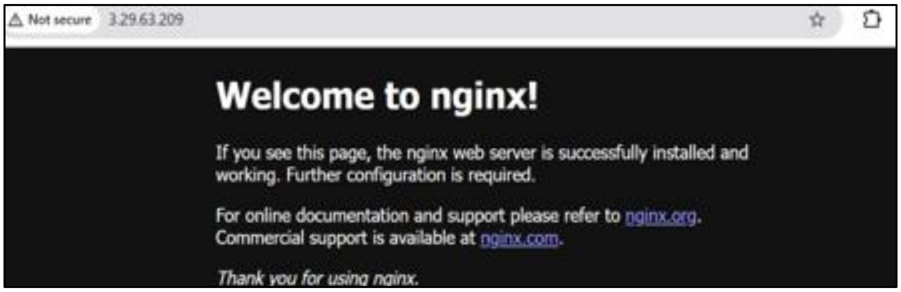
Welcome to nginx!

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.

- task6_http_redirect.png



Welcome to nginx!

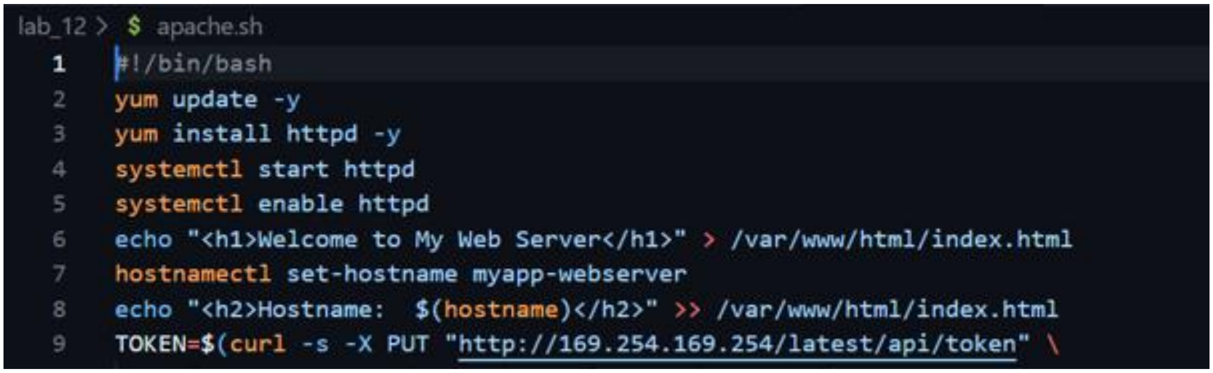
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

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

Thank you for using nginx.

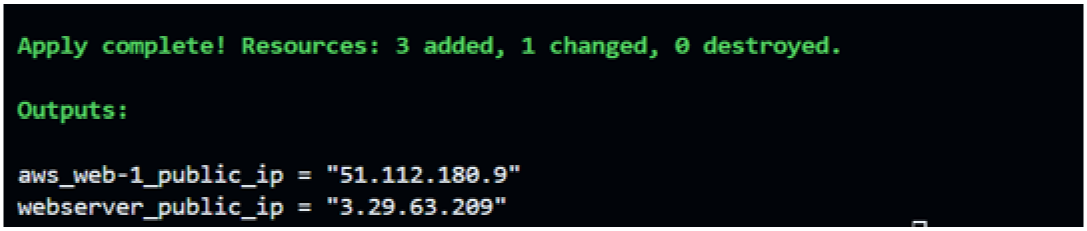
Task 7 - Configure Nginx as reverse proxy

- task7_apache_script.png



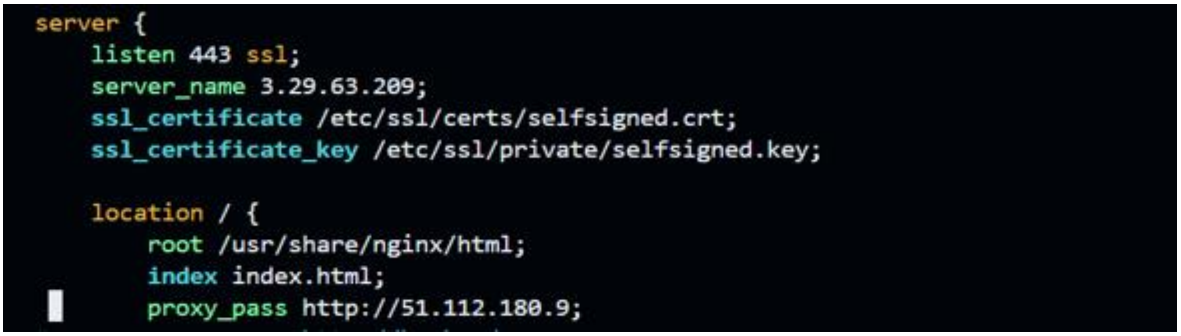
```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo "<h1>Welcome to My Web Server</h1>" > /var/www/html/index.html
hostnamectl set-hostname myapp-webserver
echo "<h2>Hostname:  $(hostname)</h2>" >> /var/www/html/index.html
TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" \
```

- task7_terraform_apply.png



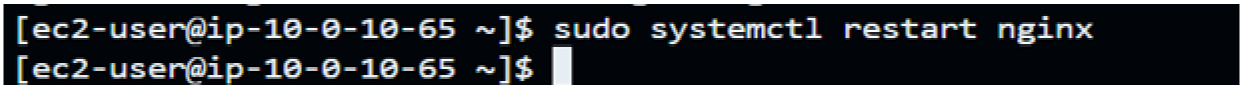
```
Apply complete! Resources: 3 added, 1 changed, 0 destroyed.

Outputs:

aws_web-1_public_ip = "51.112.180.9"
webserver_public_ip = "3.29.63.209"
```

- task7_nginx_conf_reverse_proxy.png



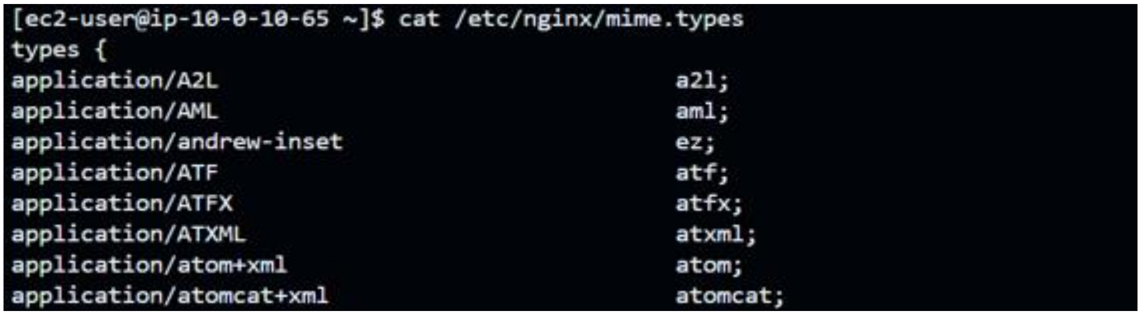
```
server {
    listen 443 ssl;
    server_name 3.29.63.209;
    ssl_certificate /etc/ssl/certs/selfsigned.crt;
    ssl_certificate_key /etc/ssl/private/selfsigned.key;

    location / {
        root /usr/share/nginx/html;
        index index.html;
        proxy_pass http://51.112.180.9;
```

- task7_nginx_restart.png



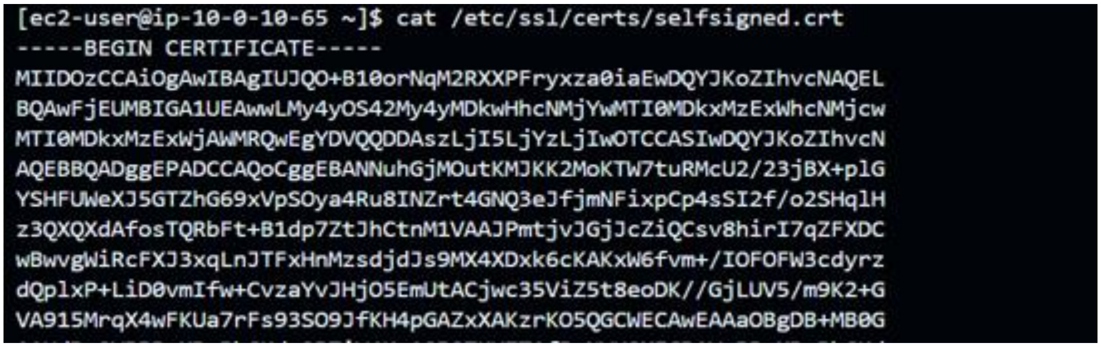
```
[ec2-user@ip-10-0-10-65 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-65 ~]$
```

- task7_mime_types.png



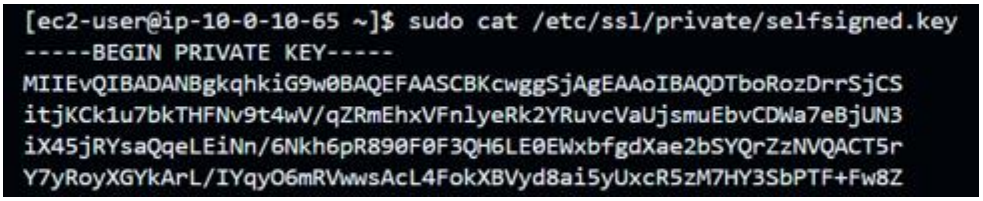
```
[ec2-user@ip-10-0-10-65 ~]$ cat /etc/nginx/mime.types
types {
application/A2L                                   a2l;
application/AML                                   aml;
application/andrew-inset                          ez;
application/ATF                                   atf;
application/ATFX                                  atfx;
application/ATXML                                 atxml;
application/atom+xml                              atom;
application/atomcat+xml                           atomcat;
```

- task7_ssl_cert.png



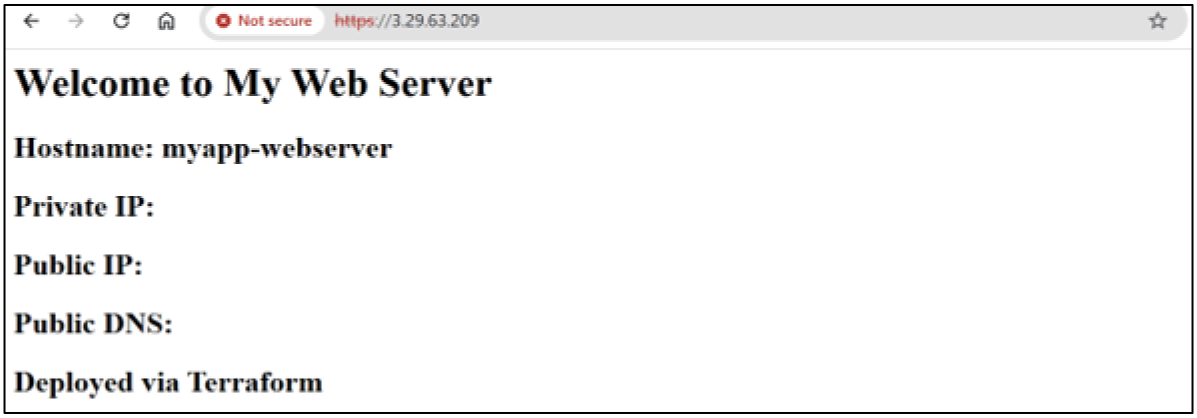
```
[ec2-user@ip-10-0-10-65 ~]$ cat /etc/ssl/certs/selfsigned.crt
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIUJQO+B10orNqM2RXXPFryxza0iaEwDQYJKoZIhvcNAQEL
BQAwFjEUMBIGA1UEAwwLMy4yOS42My4yMDkwHhcNMjYwMTI0MDkxMzExWhcNMjcw
MTI0MDkxMzExWjAWMRQwEgYDVQQDDAszLjI5LjYzLjIwOTCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBANNuhGjMOutKMJKK2MoKTW7tuRMcU2/23jBX+plG
YSHFUWeXJ5GTZhG69xVpSOya4Ru8INZrt4GNQ3eJfjmNFixpCp4sSI2f/o2SHqlH
z3QXQXdAfosTQRbFt+B1dp7ZtJhCtnM1VAAJPmtjvJGjJcZiQCsv8hirI7qZFXDC
wBwvgWiRcFXJ3xqLnJTFxHnMzsdjdJs9MX4XDxk6cKAKxW6fvm+/IOFOFW3cdyrz
dQplxP+LiD0vmIfw+CvzaYvJHjO5EmUtACjwc35ViZ5t8eoDK//GjLUV5/m9K2+G
VA915MrqX4wFKUa7rFs93SO9JfKH4pGAZxXAKzrKO5QGCWECAwEAAaOBgDB+MB0G
```

- task7_ssl_key.png



```
[ec2-user@ip-10-0-10-65 ~]$ sudo cat /etc/ssl/private/selfsigned.key
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggSjAgEAAoIBAQDTboRozDrrSjCS
itjKCk1u7bkTHFNv9t4wV/qZRmEhxVFnlyeRk2YRuvcVaUjsmuEbvCDWa7eBjUN3
iX45jRYsaQqeLEiNn/6Nkh6pR890F0F3QH6LE0EWxbfgdXae2bSYQrZzNVQACT5r
Y7yRoyXGYkArL/IYqyO6mRVwwsAcL4FokXBVyd8ai5yUxcR5zM7HY3SbPTF+Fw8Z
```

- task7_reverse_proxy_browser.png



Welcome to My Web Server

Hostname: myapp-webserver

Private IP:

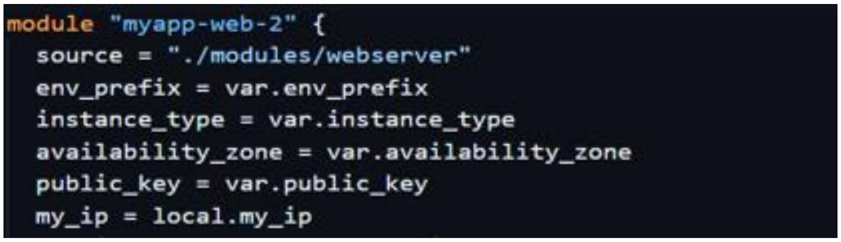
Public IP:

Public DNS:

Deployed via Terraform

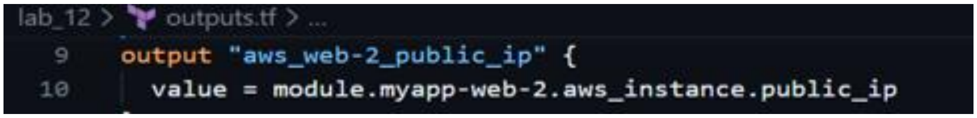
Task 8 – Configure Nginx as load balancer

- task8_main_tf_web2.png



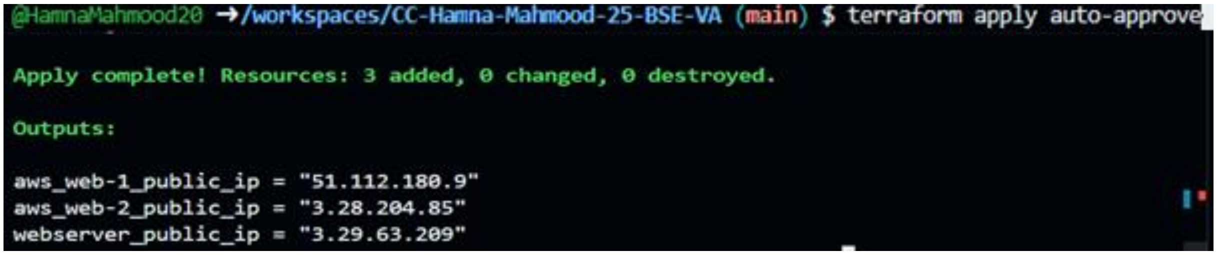
```
module "myapp-web-2" {
  source = "./modules/webserver"
  env_prefix = var.env_prefix
  instance_type = var.instance_type
  availability_zone = var.availability_zone
  public_key = var.public_key
  my_ip = local.my_ip
```

- task8_outputs_web2.png



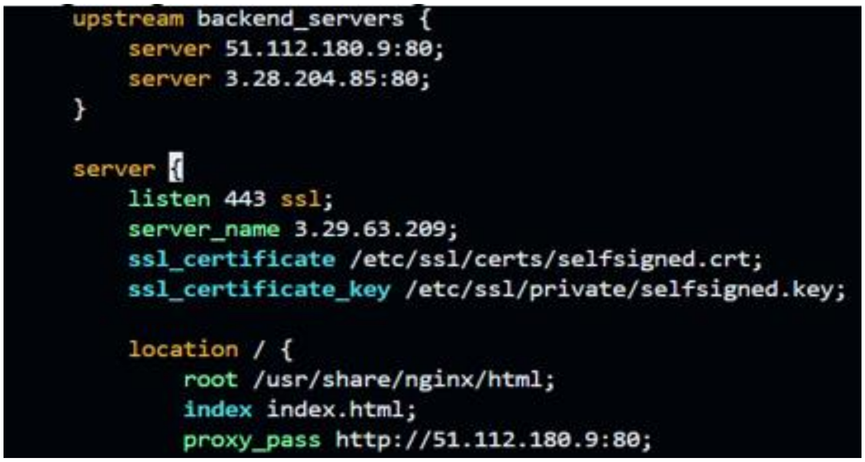
```
output "aws_web-2_public_ip" {
  value = module.myapp-web-2.aws_instance.public_ip
```

- task8_terraform_apply.png



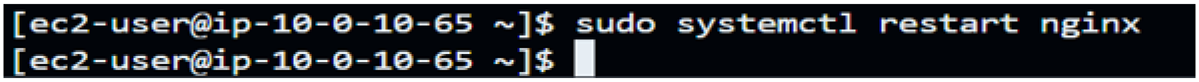
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform apply auto-approve

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

Outputs:

aws_web-1_public_ip = "51.112.180.9"
aws_web-2_public_ip = "3.28.204.85"
webserver_public_ip = "3.29.63.209"
```

- task8_nginx_conf_load_balancer.png



```
upstream backend_servers {
    server 51.112.180.9:80;
    server 3.28.204.85:80;
}

server {
    listen 443 ssl;
    server_name 3.29.63.209;
    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://51.112.180.9:80;
```

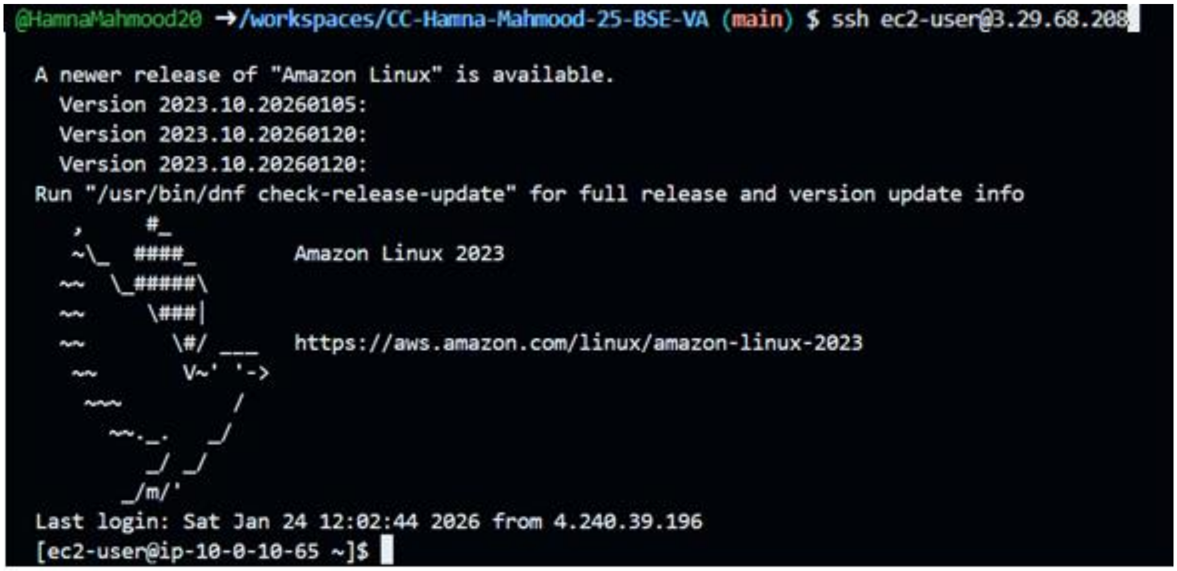
- task8_nginx_restart.png



```
[ec2-user@ip-10-0-10-65 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-65 ~]$
```

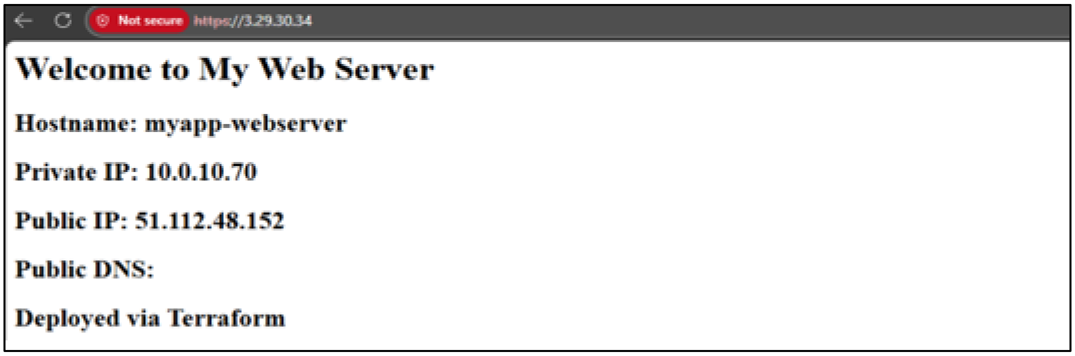
Task 9 - Configure high availability with backup servers

- task9_nginx_conf_ha_web1_primary.png



```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ ssh ec2-user@3.29.68.208

 A newer release of "Amazon Linux" is available.
   Version 2023.10.20260105:
   Version 2023.10.20260120:
   Version 2023.10.20260120:
 Run "/usr/bin/dnf check-release-update" for full release and version update info
    ,     #_
    ~\_  ####_        Amazon Linux 2023
   ~~  \_#####\
   ~~     \###|
   ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
    ~~       V~' '->
     ~~~         /
       ~~._.   _/
          _/ _/
        _/m/'
 Last login: Sat Jan 24 12:02:44 2026 from 4.240.39.196
 [ec2-user@ip-10-0-10-65 ~]$
```

- task9_ha_web1_only.png



Not secure https://3.29.30.34

Welcome to My Web Server

Hostname: myapp-webserver

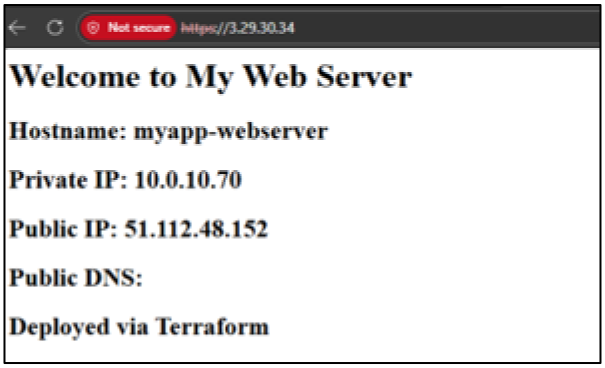
Private IP: 10.0.10.70

Public IP: 51.112.48.152

Public DNS:

Deployed via Terraform

- task9_ha_web2_only.png



Not secure https://3.29.30.34

Welcome to My Web Server

Hostname: myapp-webserver

Private IP: 10.0.10.70

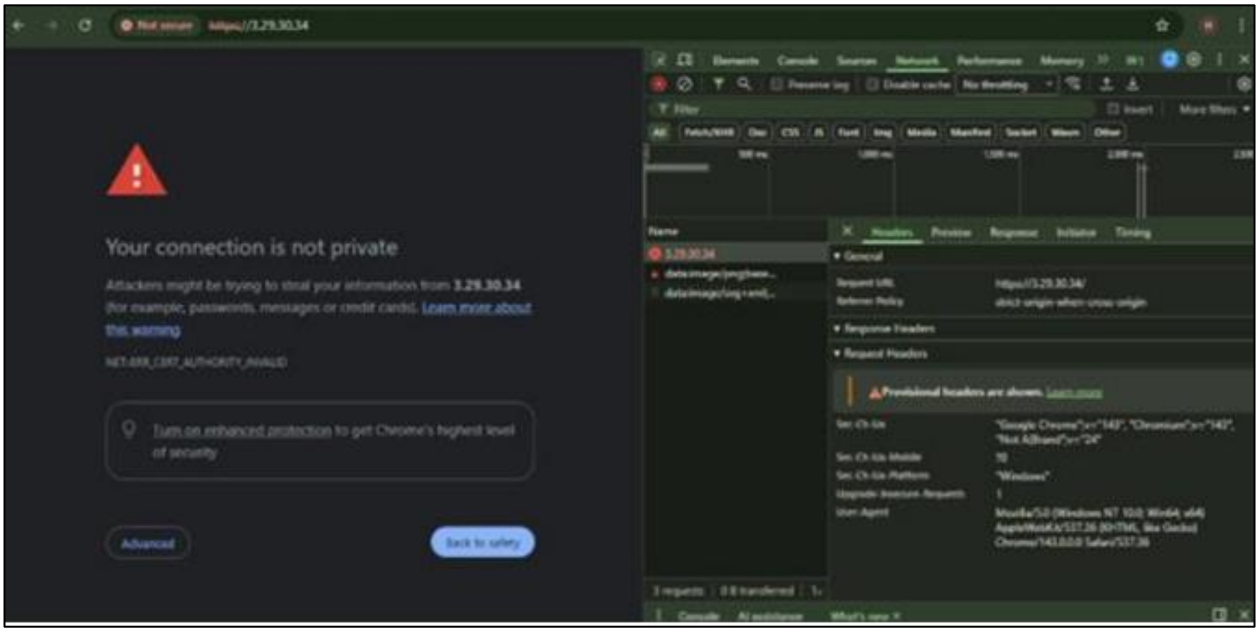
Public IP: 51.112.48.152

Public DNS:

Deployed via Terraform

Task 10 - Enable Nginx caching

- task10_nginx_conf_cache.png



Not secure https://3.29.30.34

Your connection is not private

Attackers might be trying to steal your information from **3.29.30.34** (for example, passwords, messages or credit cards). Learn more about this warning

NET::ERR_CERT_AUTHORITY_INVALID

Turn on enhanced protection to get Chrome's highest level of security

3.29.30.34
data:image/png;base...
data:image/svg+xml;...

▾ General

Request URL	https://3.29.30.34/
Referrer Policy	strict-origin-when-cross-origin

▾ Response Headers

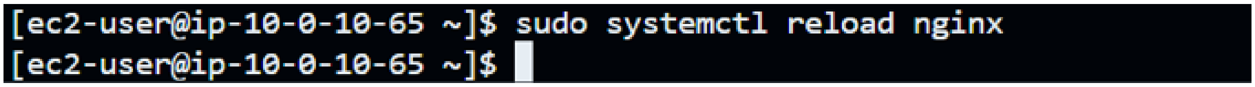
▾ Request Headers

⚠ Provisional headers are shown. Learn more

Sec-Ch-Ua	"Google Chrome";v="143", "Chromium";v="143", "Not A(Brand";v="24"
Sec-Ch-Ua-Mobile	?0
Sec-Ch-Ua-Platform	"Windows"
Upgrade-Insecure-Requests	1
User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36

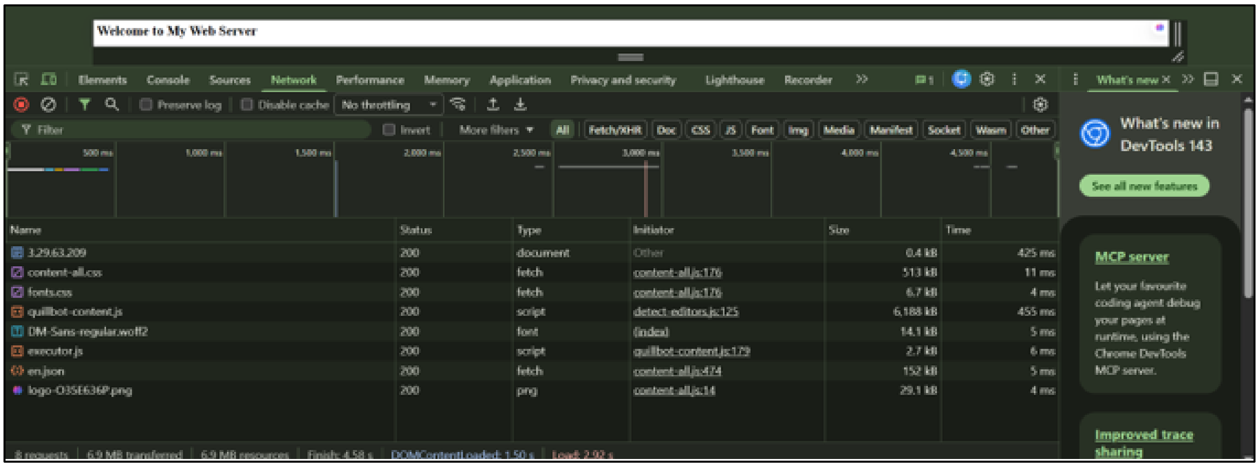
3 requests | 0 B transferred

- task10_nginx_restart.png



```
[ec2-user@ip-10-0-10-65 ~]$ sudo systemctl reload nginx
[ec2-user@ip-10-0-10-65 ~]$ 
```

- task10_cache_miss.png



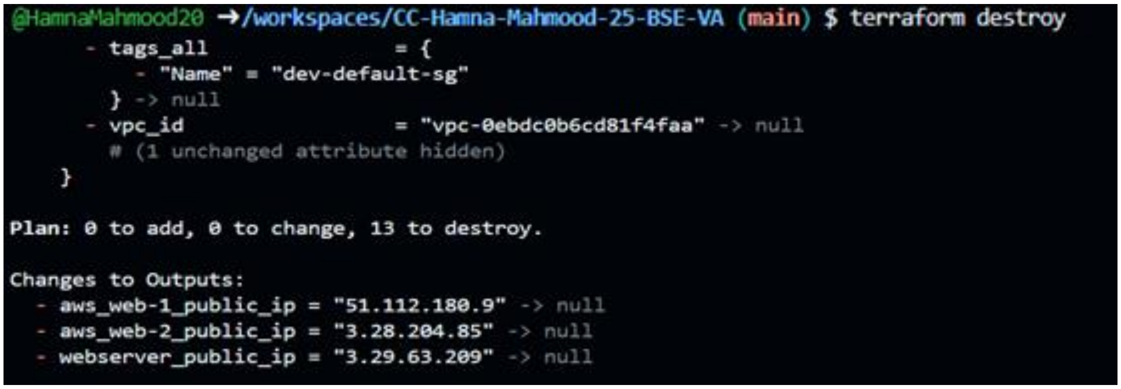
Welcome to My Web Server

Name	Status	Type	Initiator	Size	Time
3.29.63.209	200	document	Other	0.4 kB	425 ms
content-all.css	200	fetch	content-all.js:176	513 kB	11 ms
fonts.css	200	fetch	content-all.js:176	6.7 kB	4 ms
quillbot-content.js	200	script	detect-editors.js:125	6,188 kB	455 ms
DM-Sans-regular.woff2	200	font	(index)	14.1 kB	5 ms
executor.js	200	script	quillbot-content.js:179	2.7 kB	6 ms
en.json	200	fetch	content-all.js:474	152 kB	5 ms
logo-O35E636P.png	200	png	content-all.js:14	29.1 kB	4 ms

8 requests | 6.9 MB transferred | 6.9 MB resources | Finish: 4.58 s | DOMContentLoaded: 1.50 s | Load: 2.92 s

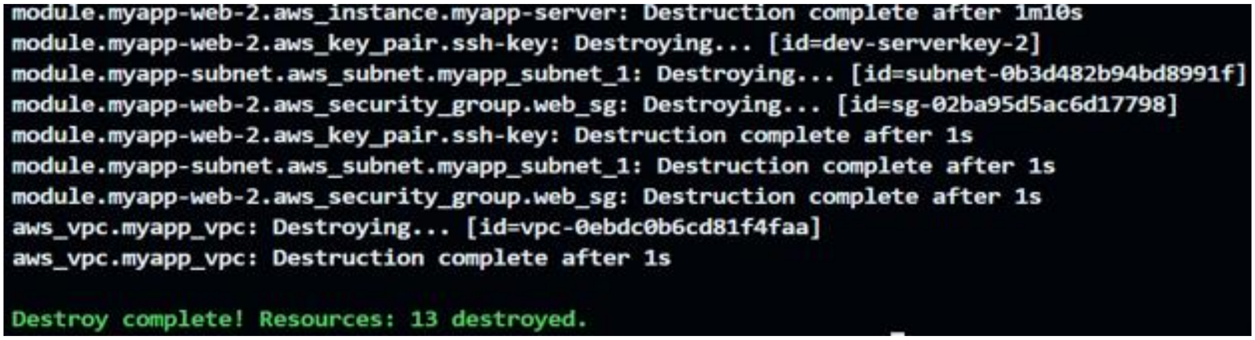
Cleanup

- cleanup_destroy_prompt.png



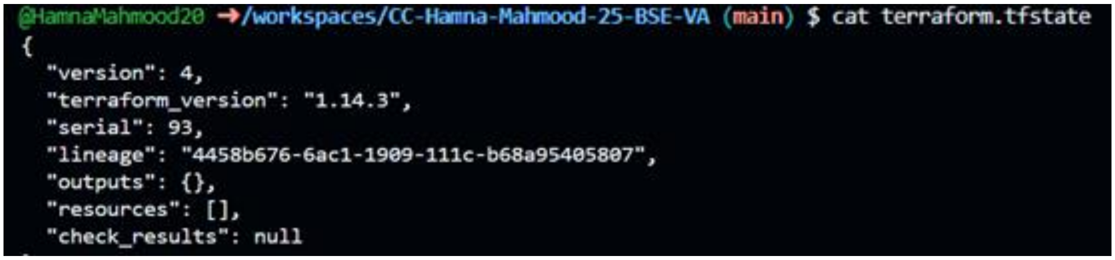
```
@HamnaMahmood20 →/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform destroy
      - tags_all               = {
          - "Name" = "dev-default-sg"
        } -> null
      - vpc_id                 = "vpc-0ebdc0b6cd81f4faa" -> null
        # (1 unchanged attribute hidden)
    }

Plan: 0 to add, 0 to change, 13 to destroy.

Changes to Outputs:
  - aws_web-1_public_ip = "51.112.180.9" -> null
  - aws_web-2_public_ip = "3.28.204.85" -> null
  - webserver_public_ip = "3.29.63.209" -> null
```

- cleanup_destroy_complete.png



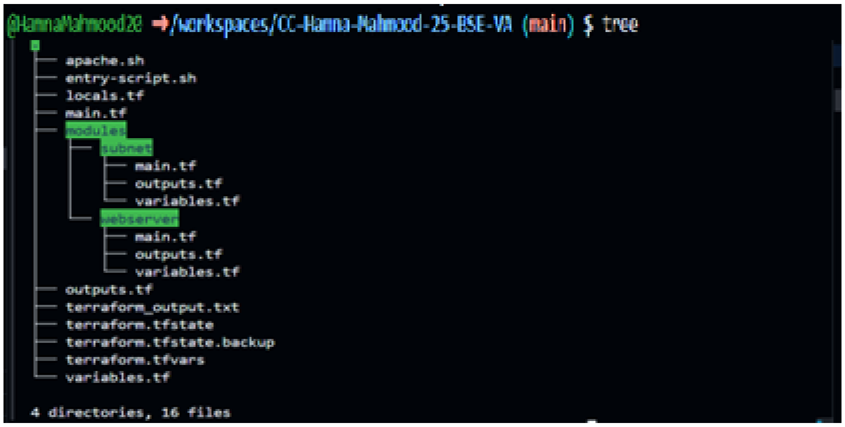
```
module.myapp-web-2.aws_instance.myapp-server: Destruction complete after 1m10s
module.myapp-web-2.aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-2]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id=subnet-0b3d482b94bd8991f]
module.myapp-web-2.aws_security_group.web_sg: Destroying... [id=sg-02ba95d5ac6d17798]
module.myapp-web-2.aws_key_pair.ssh-key: Destruction complete after 1s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 1s
module.myapp-web-2.aws_security_group.web_sg: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-0ebdc0b6cd81f4faa]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 13 destroyed.
```

- cleanup_state_empty.png



```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 93,
  "lineage": "4458b676-6ac1-1909-111c-b68a95405807",
  "outputs": {},
  "resources": [],
  "check_results": null
```

- cleanup_final_files.png



```
@HamnaMahmood20 ➜ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ tree
.
├── apache.sh
├── entry-script.sh
├── locals.tf
├── main.tf
├── modules
│   ├── subnet
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   └── webserver
│       ├── main.tf
│       ├── outputs.tf
│       └── variables.tf
├── outputs.tf
├── terraform_output.txt
├── terraform.tfstate
├── terraform.tfstate.backup
├── terraform.tfvars
└── variables.tf

4 directories, 16 files
```