FATIMA JINNAH WOMEN UNIVERSITY

Department of Software Engineering

ASSIGNMENT#02

Report

SUBJECT: CLOUD COMPUTING

SUBMITTED TO: SIR WAQAS SALEEM

SUBMITTED BY: UMBER QASIM

SUBMITTED DATE: 31 DECEMBER 2025

REGISTRATION NO: 2023-BSE-066

CLASS: BSSE V-B

Contents

Executive Summary

This assignment focuses on designing and deploying a secure, highly available, and scalable multi-tier web infrastructure on Amazon Web Services (AWS) using Terraform.

The infrastructure consists of:

- A Nginx reverse proxy/load balancer
- Multiple Apache backend web servers
- A secure VPC-based network
- Automated server configuration scripts
- Caching, SSL, security headers, rate limiting, and health checks

Terraform was used to provision all cloud resources in an Infrastructure as Code (IaC) approach, ensuring repeatability and consistency.

The system was tested for load balancing, fault tolerance, security, and performance.

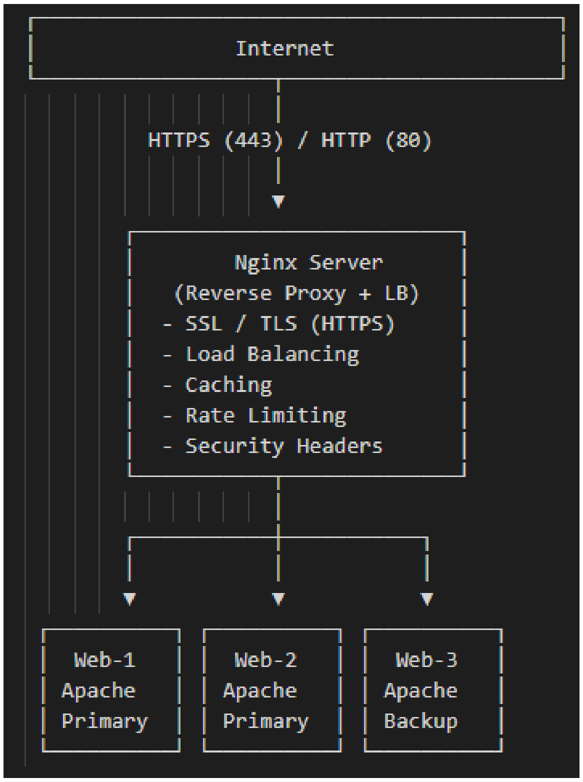
Key Achievements:

- Fully automated AWS infrastructure using Terraform
- Nginx HTTPS load balancer with SSL
- Multiple Apache backend servers with dynamic metadata display
- Secure networking using security groups
- Bonus features: rate limiting, health checks, caching, and security headers

Architecture Design

Architecture Overview

The architecture follows a three-tier design where traffic from the internet first reaches the Nginx load balancer, which then distributes requests to backend Apache servers.



```
┌──────────────────────────────────────────┐
│                 Internet                 │
└──────────────────────────────────────────┘
                     │
                     │
           HTTPS (443) / HTTP (80)
                     │
                     ▼
        ┌──────────────────────────┐
        │       Nginx Server       │
        │  (Reverse Proxy + LB)    │
        │  - SSL / TLS (HTTPS)     │
        │  - Load Balancing        │
        │  - Caching               │
        │  - Rate Limiting         │
        │  - Security Headers      │
        └──────────────────────────┘
                     │
        ┌────────────┼────────────┐
        │            │            │
        ▼            ▼            ▼
 ┌──────────┐  ┌──────────┐  ┌──────────┐
 │  Web-1   │  │  Web-2   │  │  Web-3   │
 │ Apache   │  │ Apache   │  │ Apache   │
 │ Primary  │  │ Primary  │  │ Backup   │
 └──────────┘  └──────────┘  └──────────┘
```

Component Descriptions

Nginx Server

- Acts as a reverse proxy and load balancer
- Terminates SSL (HTTPS)
- Forwards requests to backend servers
- Implements caching and rate limiting
- Adds security headers

Backend Web Servers (Apache)

- Serve dynamic HTML pages
- Display instance metadata (IP, hostname, timestamp)
- Identified as web-1, web-2, and web-3
- web-3 acts as a backup server

Networking Components

- VPC
- Subnet

- ❖ Internet Gateway
- ❖ Route Table

Network Topology

- ❖ All servers are deployed inside a single VPC
- ❖ Public subnet allows internet access
- ❖ Security groups restrict traffic between layers
- ❖ Backend servers only accept traffic from Nginx

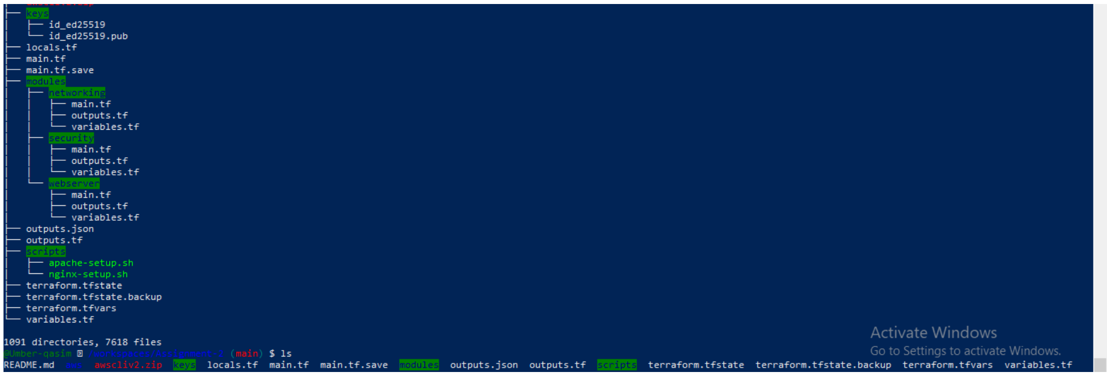
Security Design

- ❖ SSH access restricted to personal IP
- ❖ Backend servers not publicly exposed
- ❖ HTTPS encryption using SSL certificates
- ❖ Security headers enabled
- ❖ Rate limiting to prevent abuse

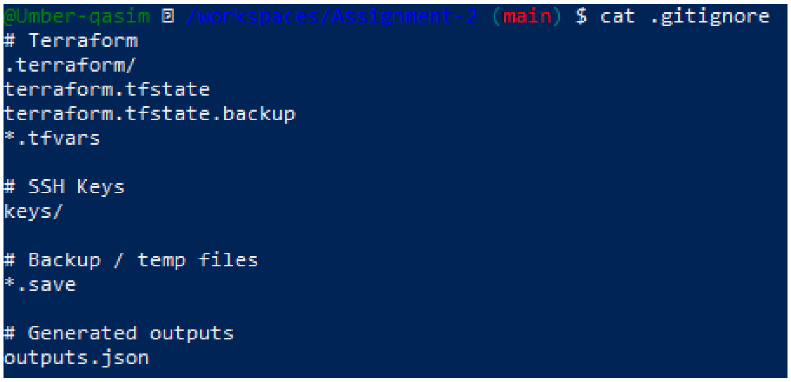
Implementation Details

Part 1: Infrastructure Setup

1.1 Project Structure



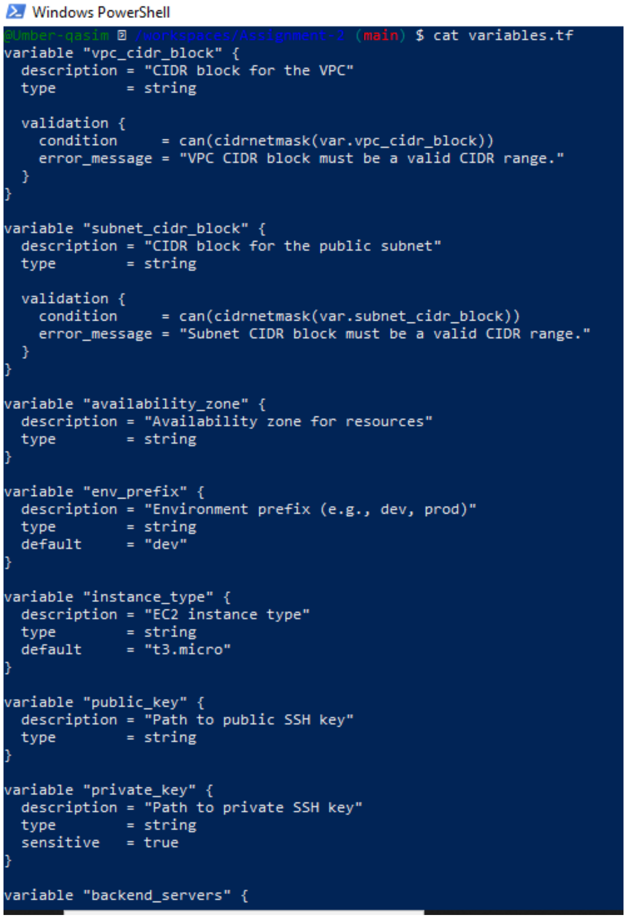
```
├── keys
│   ├── id_ed25519
│   └── id_ed25519.pub
├── locals.tf
├── main.tf
├── main.tf.save
├── modules
│   ├── networking
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   ├── security
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   └── webserver
│       ├── main.tf
│       ├── outputs.tf
│       └── variables.tf
├── outputs.json
├── outputs.tf
├── scripts
│   ├── apache-setup.sh
│   └── nginx-setup.sh
├── terraform.tfstate
├── terraform.tfstate.backup
├── terraform.tfvars
└── variables.tf

1091 directories, 7618 files
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $ ls
README.md  aws  awscliv2.zip  keys  locals.tf  main.tf  main.tf.save  modules  outputs.json  outputs.tf  scripts  terraform.tfstate  terraform.tfstate.backup  terraform.tfvars  variables.tf
```



```
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $ cat .gitignore
# Terraform
.terraform/
terraform.tfstate
terraform.tfstate.backup
*.tfvars

# SSH Keys
keys/

# Backup / temp files
*.save

# Generated outputs
outputs.json
```

1.2 Variable Configuration

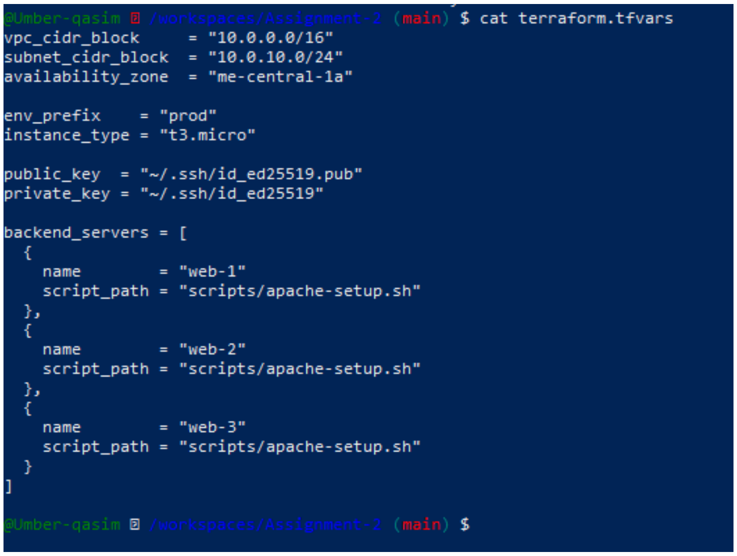
File: variables.tf



Windows PowerShell

```
@Umber-qasim ⍰ /workspaces/Assignment-2 (main) $ cat variables.tf
variable "vpc_cidr_block" {
  description = "CIDR block for the VPC"
  type        = string

  validation {
    condition     = can(cidrnetmask(var.vpc_cidr_block))
    error_message = "VPC CIDR block must be a valid CIDR range."
  }
}

variable "subnet_cidr_block" {
  description = "CIDR block for the public subnet"
  type        = string

  validation {
    condition     = can(cidrnetmask(var.subnet_cidr_block))
    error_message = "Subnet CIDR block must be a valid CIDR range."
  }
}

variable "availability_zone" {
  description = "Availability zone for resources"
  type        = string
}

variable "env_prefix" {
  description = "Environment prefix (e.g., dev, prod)"
  type        = string
  default     = "dev"
}

variable "instance_type" {
  description = "EC2 instance type"
  type        = string
  default     = "t3.micro"
}

variable "public_key" {
  description = "Path to public SSH key"
  type        = string
}

variable "private_key" {
  description = "Path to private SSH key"
  type        = string
  sensitive   = true
}

variable "backend_servers" {
```

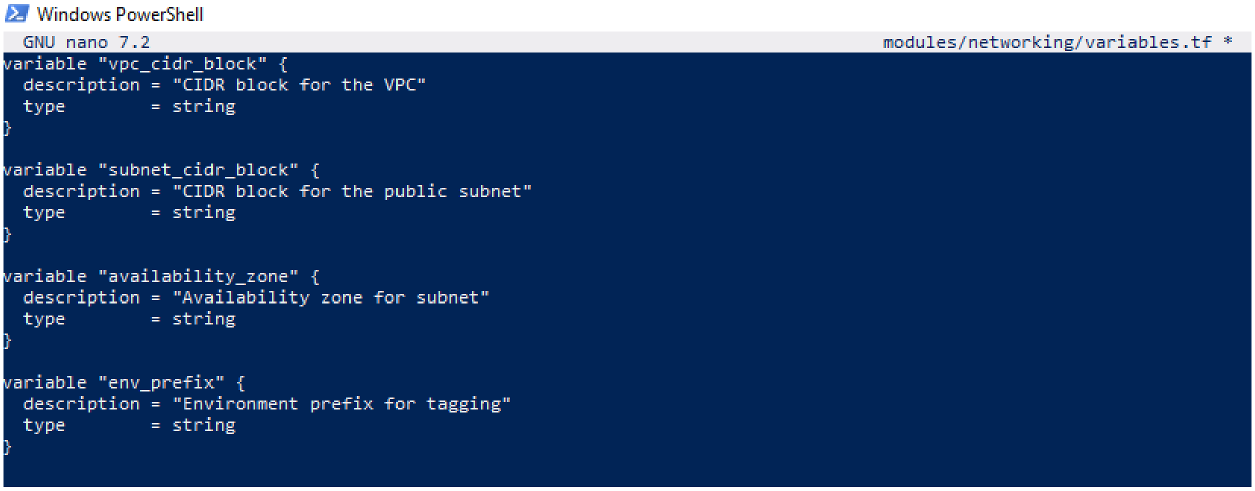
File: terraform.tfvars



```
@Umber-qasim ⍰ /workspaces/Assignment-2 (main) $ cat 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    = "prod"
instance_type = "t3.micro"

public_key  = "~/.ssh/id_ed25519.pub"
private_key = "~/.ssh/id_ed25519"

backend_servers = [
  {
    name        = "web-1"
    script_path = "scripts/apache-setup.sh"
  },
  {
    name        = "web-2"
    script_path = "scripts/apache-setup.sh"
  },
  {
    name        = "web-3"
    script_path = "scripts/apache-setup.sh"
  }
]

@Umber-qasim ⍰ /workspaces/Assignment-2 (main) $
```

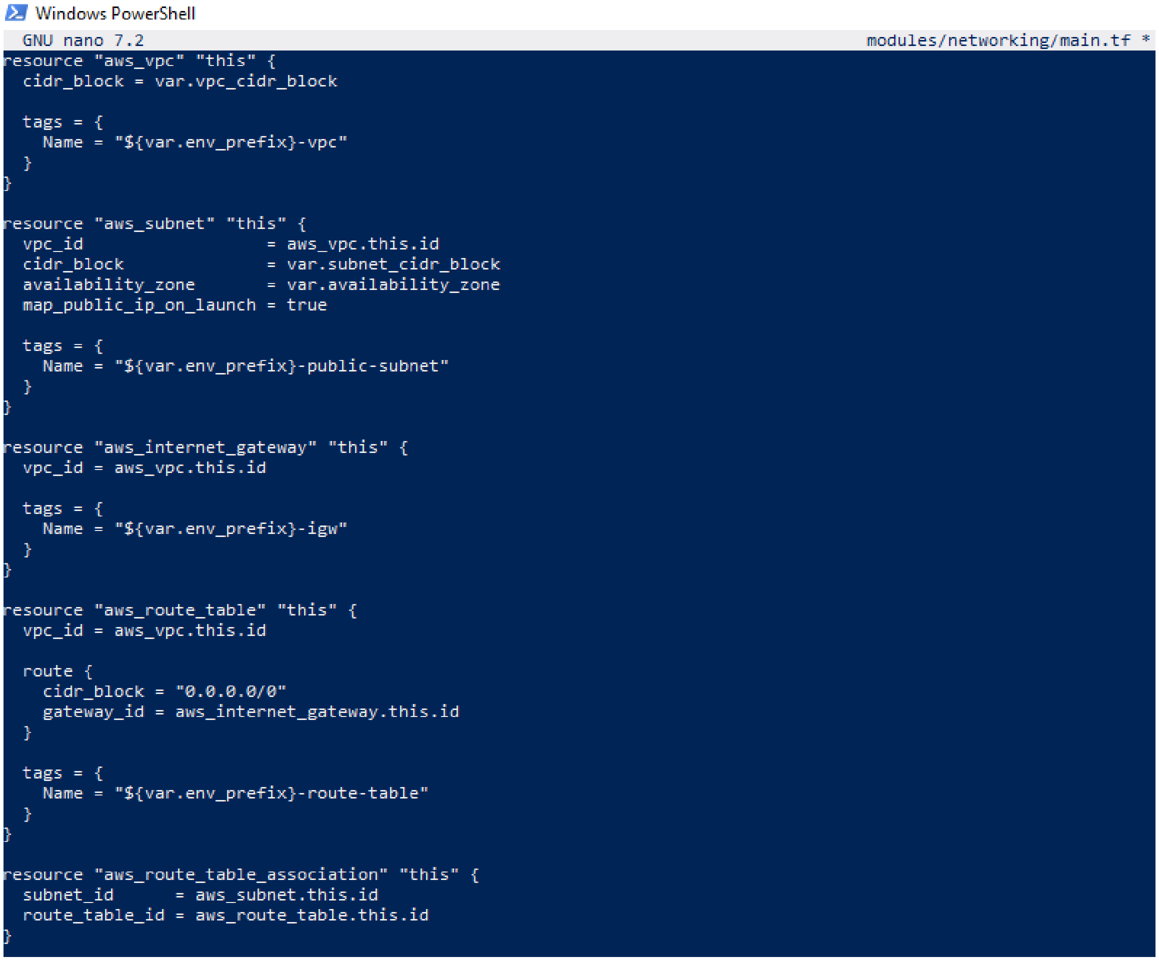
1.3 Networking Module

File: modules/networking/variables.tf



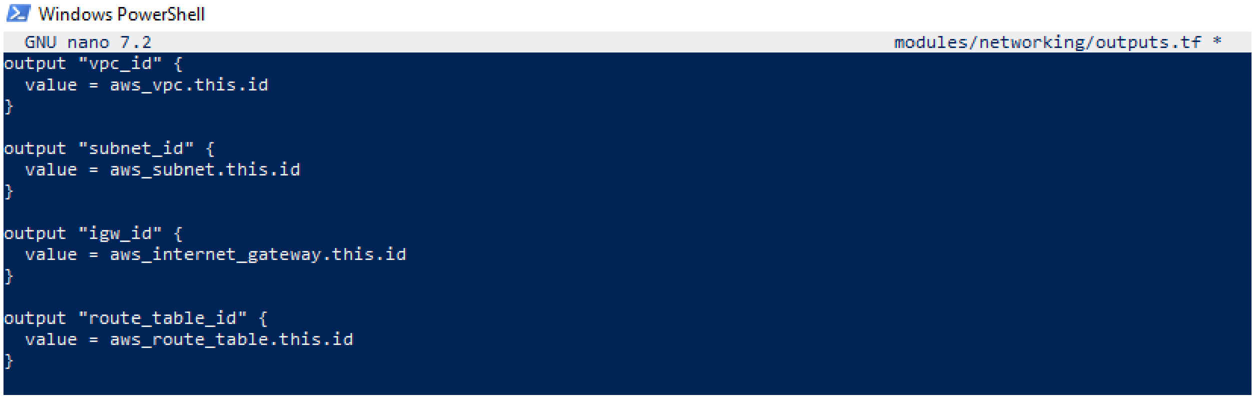
```
  GNU nano 7.2                                                           modules/networking/variables.tf *
variable "vpc_cidr_block" {
  description = "CIDR block for the VPC"
  type        = string
}

variable "subnet_cidr_block" {
  description = "CIDR block for the public subnet"
  type        = string
}

variable "availability_zone" {
  description = "Availability zone for subnet"
  type        = string
}

variable "env_prefix" {
  description = "Environment prefix for tagging"
  type        = string
}
```

File: modules/networking/main.tf



```
  GNU nano 7.2                                                     modules/networking/main.tf *
resource "aws_vpc" "this" {
  cidr_block = var.vpc_cidr_block

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

resource "aws_subnet" "this" {
  vpc_id                  = aws_vpc.this.id
  cidr_block              = var.subnet_cidr_block
  availability_zone       = var.availability_zone
  map_public_ip_on_launch = true

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

resource "aws_internet_gateway" "this" {
  vpc_id = aws_vpc.this.id

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

resource "aws_route_table" "this" {
  vpc_id = aws_vpc.this.id

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

  tags = {
    Name = "${var.env_prefix}-route-table"
  }
}

resource "aws_route_table_association" "this" {
  subnet_id      = aws_subnet.this.id
  route_table_id = aws_route_table.this.id
}
```

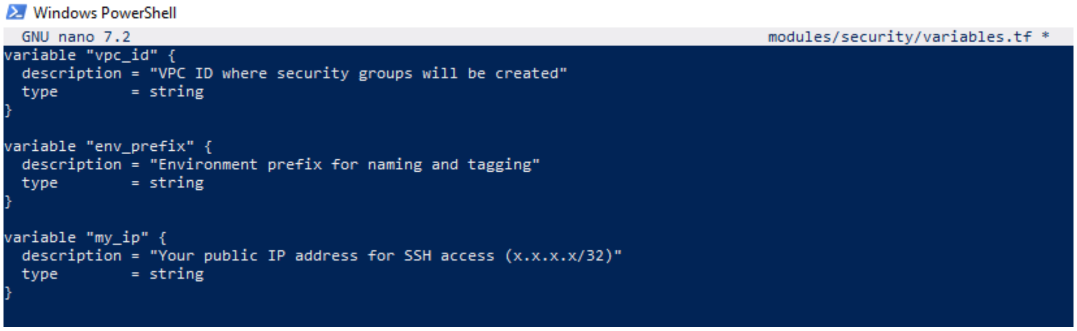
File: modules/networking/outputs.tf



```
output "vpc_id" {
  value = aws_vpc.this.id
}

output "subnet_id" {
  value = aws_subnet.this.id
}

output "igw_id" {
  value = aws_internet_gateway.this.id
}

output "route_table_id" {
  value = aws_route_table.this.id
}
```

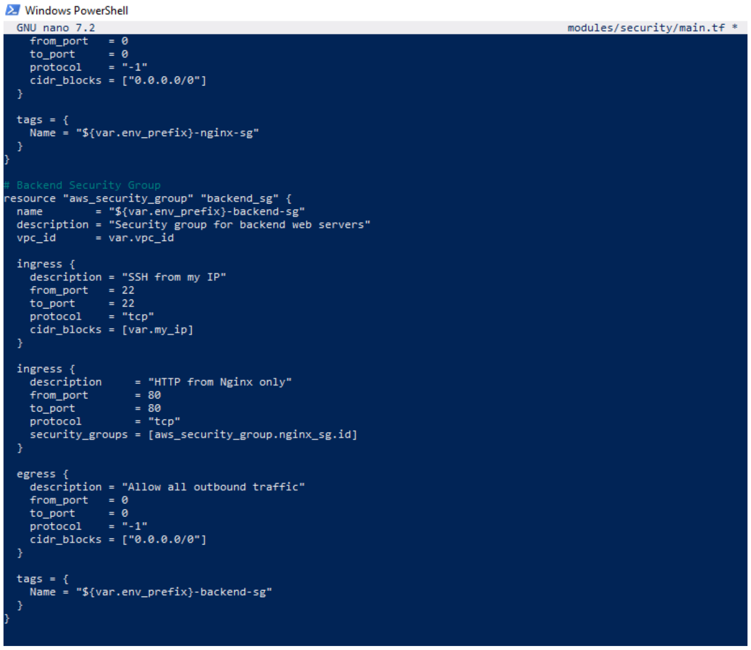
1.4 Security Module

File: modules/security/variables.tf



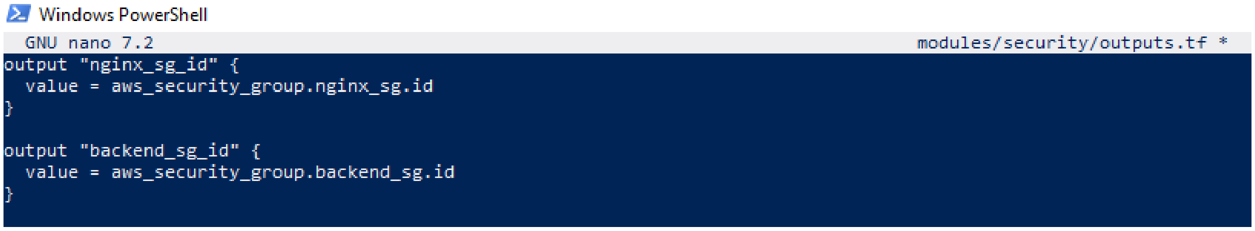
```
variable "vpc_id" {
  description = "VPC ID where security groups will be created"
  type        = string
}

variable "env_prefix" {
  description = "Environment prefix for naming and tagging"
  type        = string
}

variable "my_ip" {
  description = "Your public IP address for SSH access (x.x.x.x/32)"
  type        = string
}
```

File: modules/security/main.tf



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

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

# Backend Security Group
resource "aws_security_group" "backend_sg" {
  name        = "${var.env_prefix}-backend-sg"
  description = "Security group for backend web servers"
  vpc_id      = var.vpc_id

  ingress {
    description = "SSH from my IP"
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = [var.my_ip]
  }

  ingress {
    description     = "HTTP from Nginx only"
    from_port       = 80
    to_port         = 80
    protocol        = "tcp"
    security_groups = [aws_security_group.nginx_sg.id]
  }

  egress {
    description = "Allow all outbound traffic"
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }

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

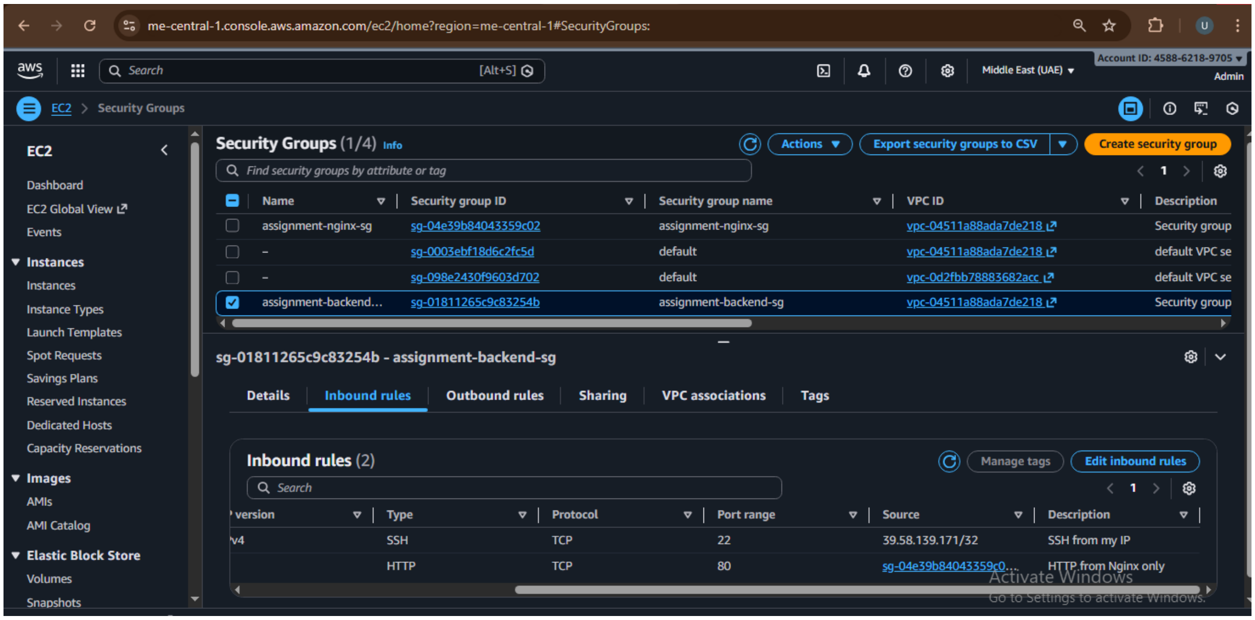
File: modules/security/outputs.tf



Windows PowerShell

```
  GNU nano 7.2                                              modules/security/outputs.tf *
output "nginx_sg_id" {
  value = aws_security_group.nginx_sg.id
}

output "backend_sg_id" {
  value = aws_security_group.backend_sg.id
}
```

AWS Console



Security Groups (1/4) Info

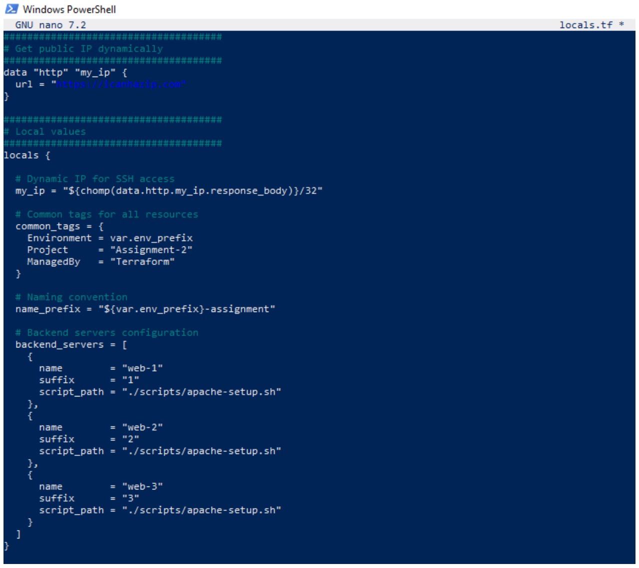
	Name	Security group ID	Security group name	VPC ID	Description
□	assignment-nginx-sg	sg-04e39b84043359c02	assignment-nginx-sg	vpc-04511a88ada7de218	Security group
□	–	sg-0003ebf18d6c2fc5d	default	vpc-04511a88ada7de218	default VPC se
□	–	sg-098e2430f9603d702	default	vpc-0d2fbb78883682acc	default VPC se
☑	assignment-backend...	sg-01811265c9c83254b	assignment-backend-sg	vpc-04511a88ada7de218	Security group

sg-01811265c9c83254b - assignment-backend-sg

Inbound rules (2)

version	Type	Protocol	Port range	Source	Description
v4	SSH	TCP	22	39.58.139.171/32	SSH from my IP
	HTTP	TCP	80	sg-04e39b84043359c0...	HTTP from Nginx only

1.5 Locals Configuration



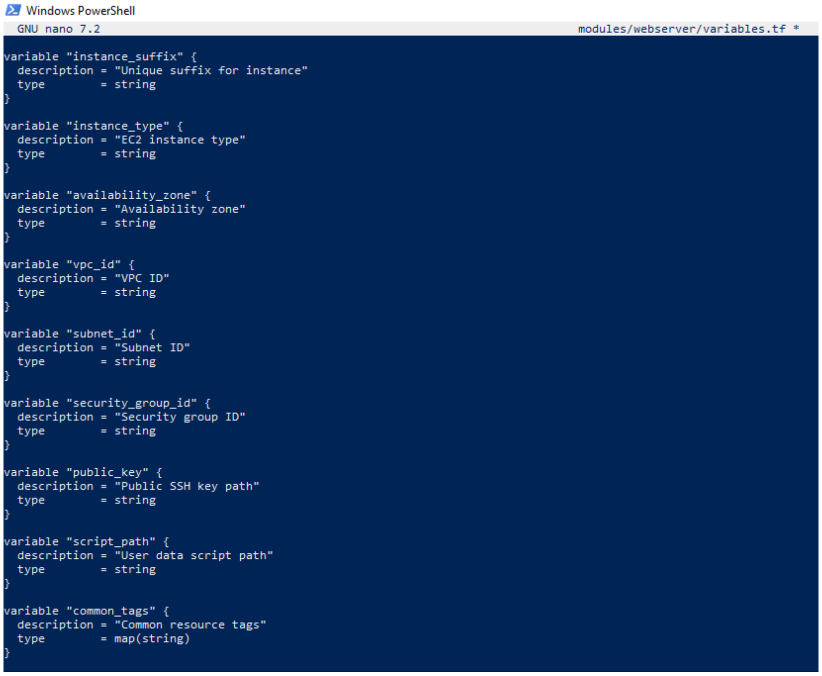
Windows PowerShell

```
  GNU nano 7.2                                          locals.tf *
#####################################
# Get public IP dynamically
#####################################
data "http" "my_ip" {
  url = "https://icanhazip.com"
}

#####################################
# Local values
#####################################
locals {

  # Dynamic IP for SSH access
  my_ip = "${chomp(data.http.my_ip.response_body)}/32"

  # Common tags for all resources
  common_tags = {
    Environment = var.env_prefix
    Project     = "Assignment-2"
    ManagedBy   = "Terraform"
  }

  # Naming convention
  name_prefix = "${var.env_prefix}-assignment"

  # Backend servers configuration
  backend_servers = [
    {
      name        = "web-1"
      suffix      = "1"
      script_path = "./scripts/apache-setup.sh"
    },
    {
      name        = "web-2"
      suffix      = "2"
      script_path = "./scripts/apache-setup.sh"
    },
    {
      name        = "web-3"
      suffix      = "3"
      script_path = "./scripts/apache-setup.sh"
    }
  ]
}
```

Part 2: Webserver Module

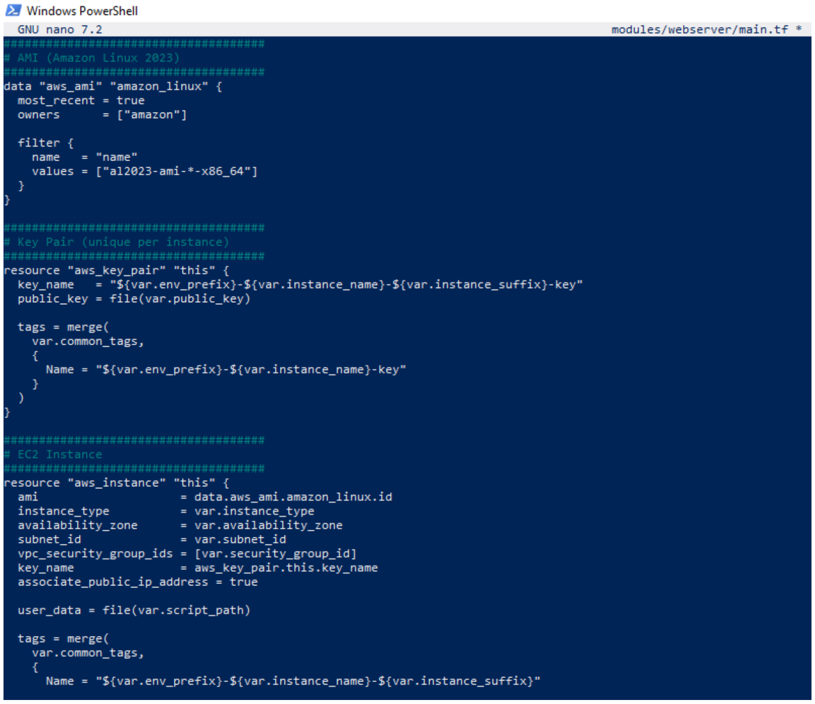
2.1 Module Design

File: modules/webserver/variables.tf



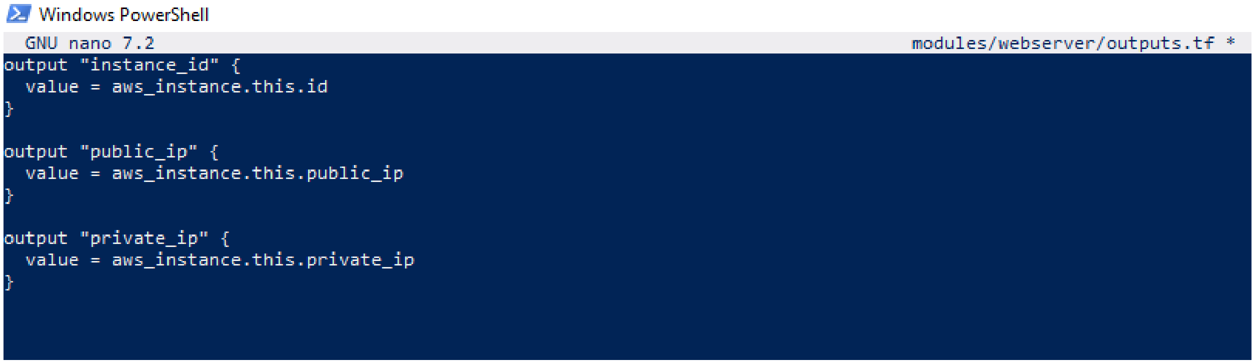
```
variable "instance_suffix" {
  description = "Unique suffix for instance"
  type        = string
}

variable "instance_type" {
  description = "EC2 instance type"
  type        = string
}

variable "availability_zone" {
  description = "Availability zone"
  type        = string
}

variable "vpc_id" {
  description = "VPC ID"
  type        = string
}

variable "subnet_id" {
  description = "Subnet ID"
  type        = string
}

variable "security_group_id" {
  description = "Security group ID"
  type        = string
}

variable "public_key" {
  description = "Public SSH key path"
  type        = string
}

variable "script_path" {
  description = "User data script path"
  type        = string
}

variable "common_tags" {
  description = "Common resource tags"
  type        = map(string)
}
```

File: modules/webserver/main.tf



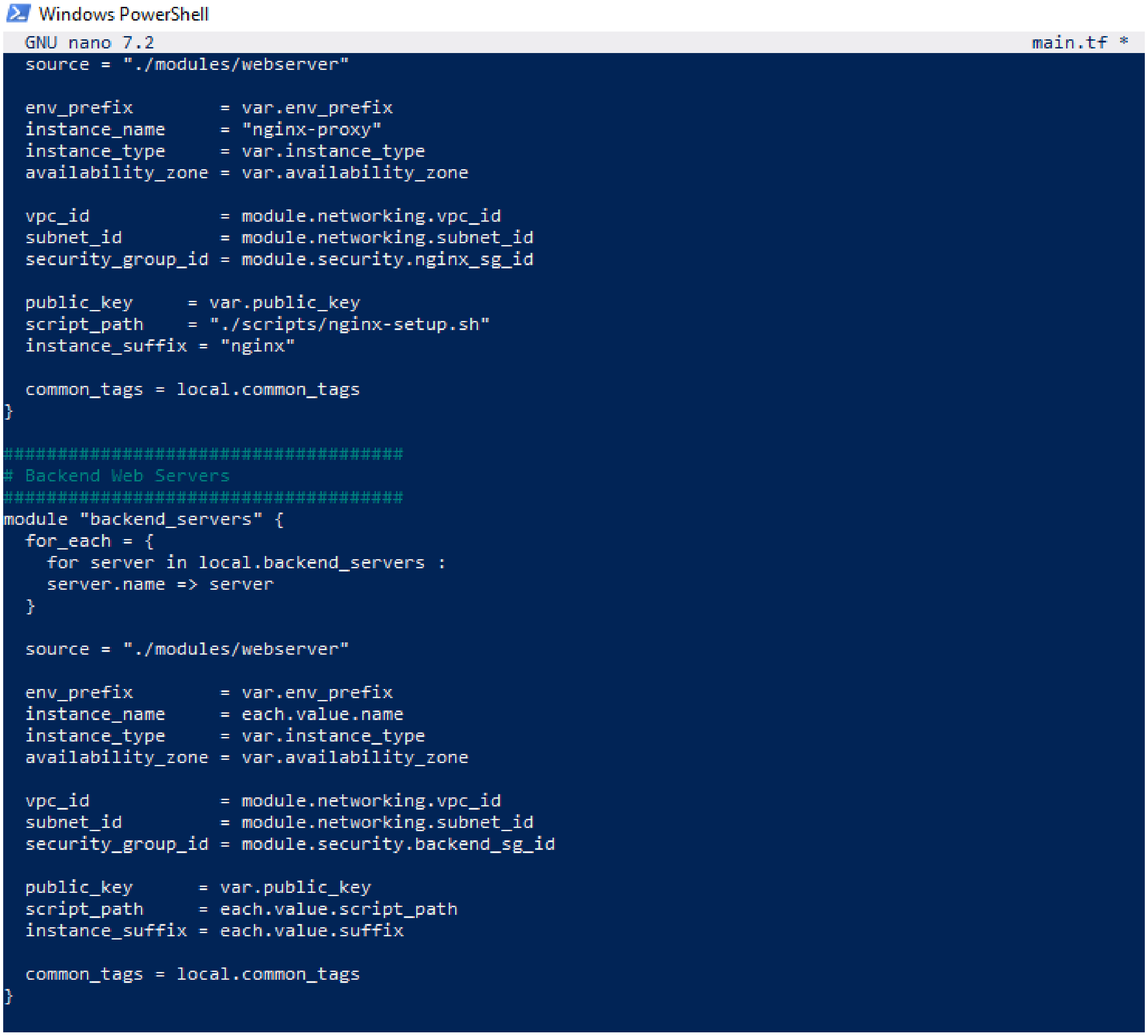
```
#####################################
# AMI (Amazon Linux 2023)
#####################################
data "aws_ami" "amazon_linux" {
  most_recent = true
  owners      = ["amazon"]

  filter {
    name   = "name"
    values = ["al2023-ami-*-x86_64"]
  }
}

#####################################
# Key Pair (unique per instance)
#####################################
resource "aws_key_pair" "this" {
  key_name   = "${var.env_prefix}-${var.instance_name}-${var.instance_suffix}-key"
  public_key = file(var.public_key)

  tags = merge(
    var.common_tags,
    {
      Name = "${var.env_prefix}-${var.instance_name}-key"
    }
  )
}

#####################################
# EC2 Instance
#####################################
resource "aws_instance" "this" {
  ami                    = data.aws_ami.amazon_linux.id
  instance_type          = var.instance_type
  availability_zone      = var.availability_zone
  subnet_id              = var.subnet_id
  vpc_security_group_ids = [var.security_group_id]
  key_name               = aws_key_pair.this.key_name
  associate_public_ip_address = true

  user_data = file(var.script_path)

  tags = merge(
    var.common_tags,
    {
      Name = "${var.env_prefix}-${var.instance_name}-${var.instance_suffix}"
```

File: modules/webserver/outputs.tf



```
output "instance_id" {
  value = aws_instance.this.id
}

output "public_ip" {
  value = aws_instance.this.public_ip
}

output "private_ip" {
  value = aws_instance.this.private_ip
}
```

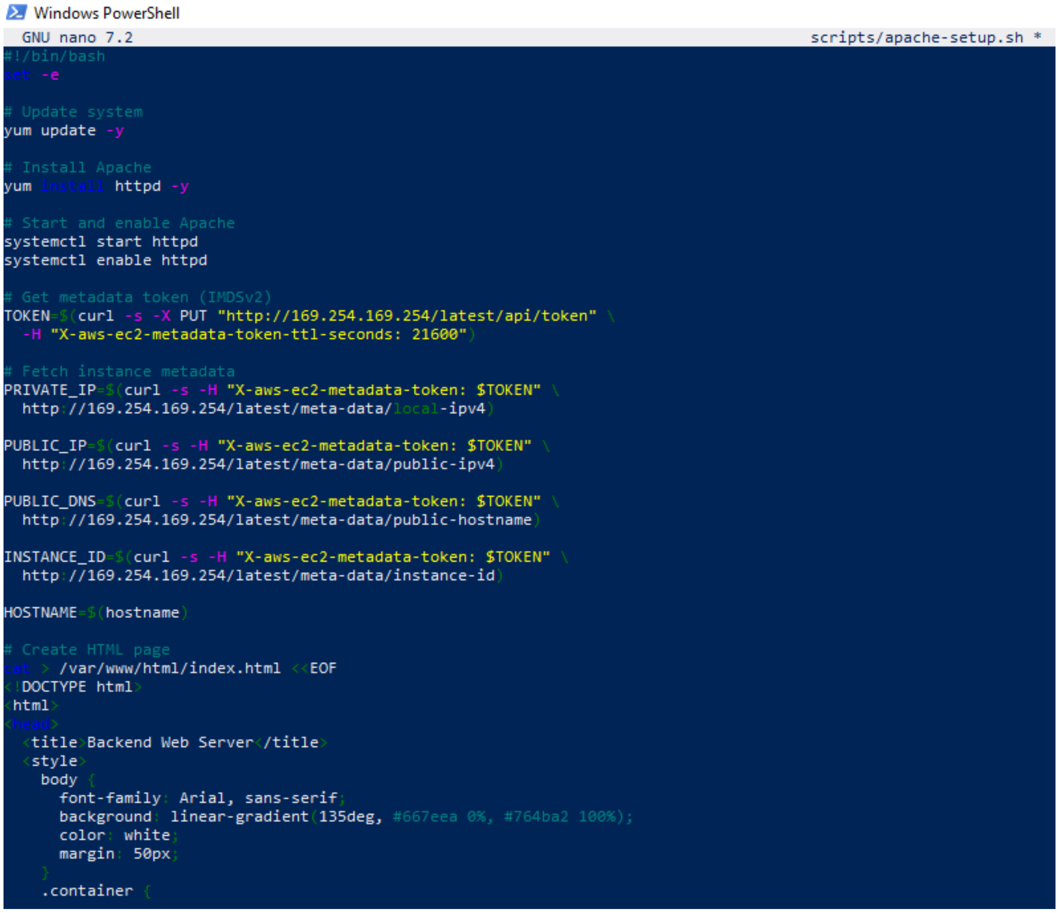
2.2 Module Usage



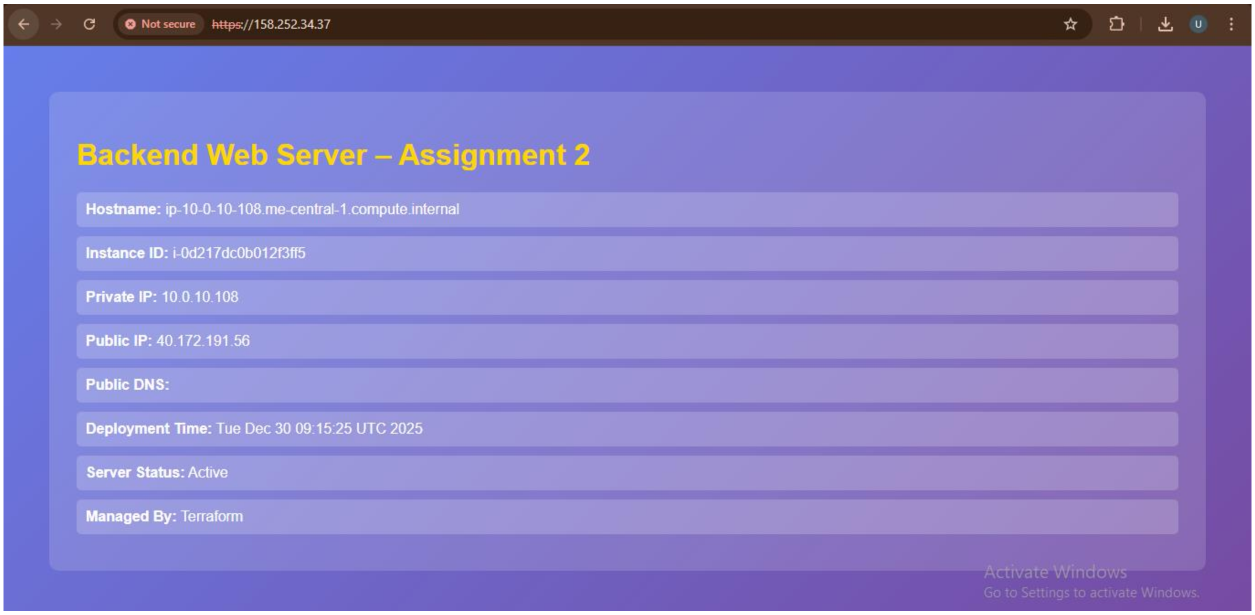
```
  source = "./modules/webserver"

  env_prefix        = var.env_prefix
  instance_name     = "nginx-proxy"
  instance_type     = var.instance_type
  availability_zone = var.availability_zone

  vpc_id            = module.networking.vpc_id
  subnet_id         = module.networking.subnet_id
  security_group_id = module.security.nginx_sg_id

  public_key     = var.public_key
  script_path    = "./scripts/nginx-setup.sh"
  instance_suffix = "nginx"

  common_tags = local.common_tags
}

#####################################
# Backend Web Servers
#####################################
module "backend_servers" {
  for_each = {
    for server in local.backend_servers :
    server.name => server
  }

  source = "./modules/webserver"

  env_prefix        = var.env_prefix
  instance_name     = each.value.name
  instance_type     = var.instance_type
  availability_zone = var.availability_zone

  vpc_id            = module.networking.vpc_id
  subnet_id         = module.networking.subnet_id
  security_group_id = module.security.backend_sg_id

  public_key      = var.public_key
  script_path     = each.value.script_path
  instance_suffix = each.value.suffix

  common_tags = local.common_tags
}
```

Part 3: Server Configuration Scripts

3.1 Apache Backend Server Script



```
  GNU nano 7.2                                                scripts/apache-setup.sh *
#!/bin/bash
set -e

# Update system
yum update -y

# Install Apache
yum install httpd -y

# Start and enable Apache
systemctl start httpd
systemctl enable httpd

# Get metadata token (IMDSv2)
TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" \
  -H "X-aws-ec2-metadata-token-ttl-seconds: 21600")

# Fetch instance metadata
PRIVATE_IP=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" \
  http://169.254.169.254/latest/meta-data/local-ipv4)

PUBLIC_IP=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" \
  http://169.254.169.254/latest/meta-data/public-ipv4)

PUBLIC_DNS=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" \
  http://169.254.169.254/latest/meta-data/public-hostname)

INSTANCE_ID=$(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" \
  http://169.254.169.254/latest/meta-data/instance-id)

HOSTNAME=$(hostname)

# Create HTML page
cat > /var/www/html/index.html <<EOF
<!DOCTYPE html>
<html>
<head>
  <title>Backend Web Server</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
      color: white;
      margin: 50px;
    }
    .container {
```



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-108.me-central-1.compute.internal

Instance ID: i-0d217dc0b012f3ff5

Private IP: 10.0.10.108

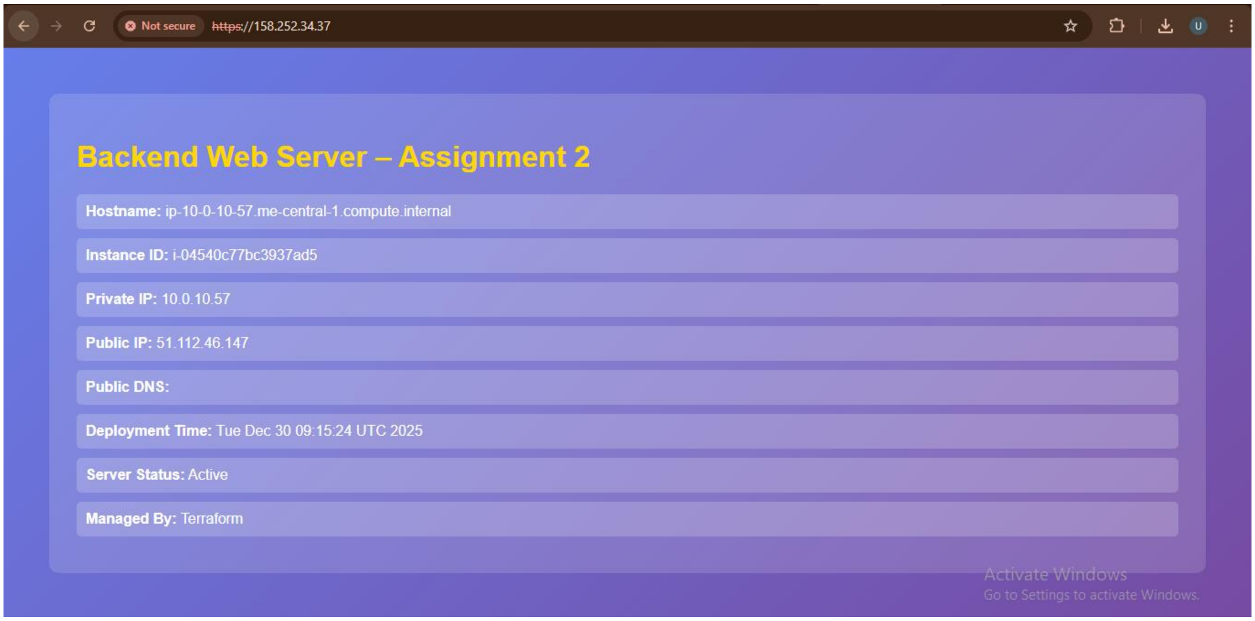
Public IP: 40.172.191.56

Public DNS:

Deployment Time: Tue Dec 30 09:15:25 UTC 2025

Server Status: Active

Managed By: Terraform



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-57.me-central-1.compute.internal

Instance ID: i-04540c77bc3937ad5

Private IP: 10.0.10.57

Public IP: 51.112.46.147

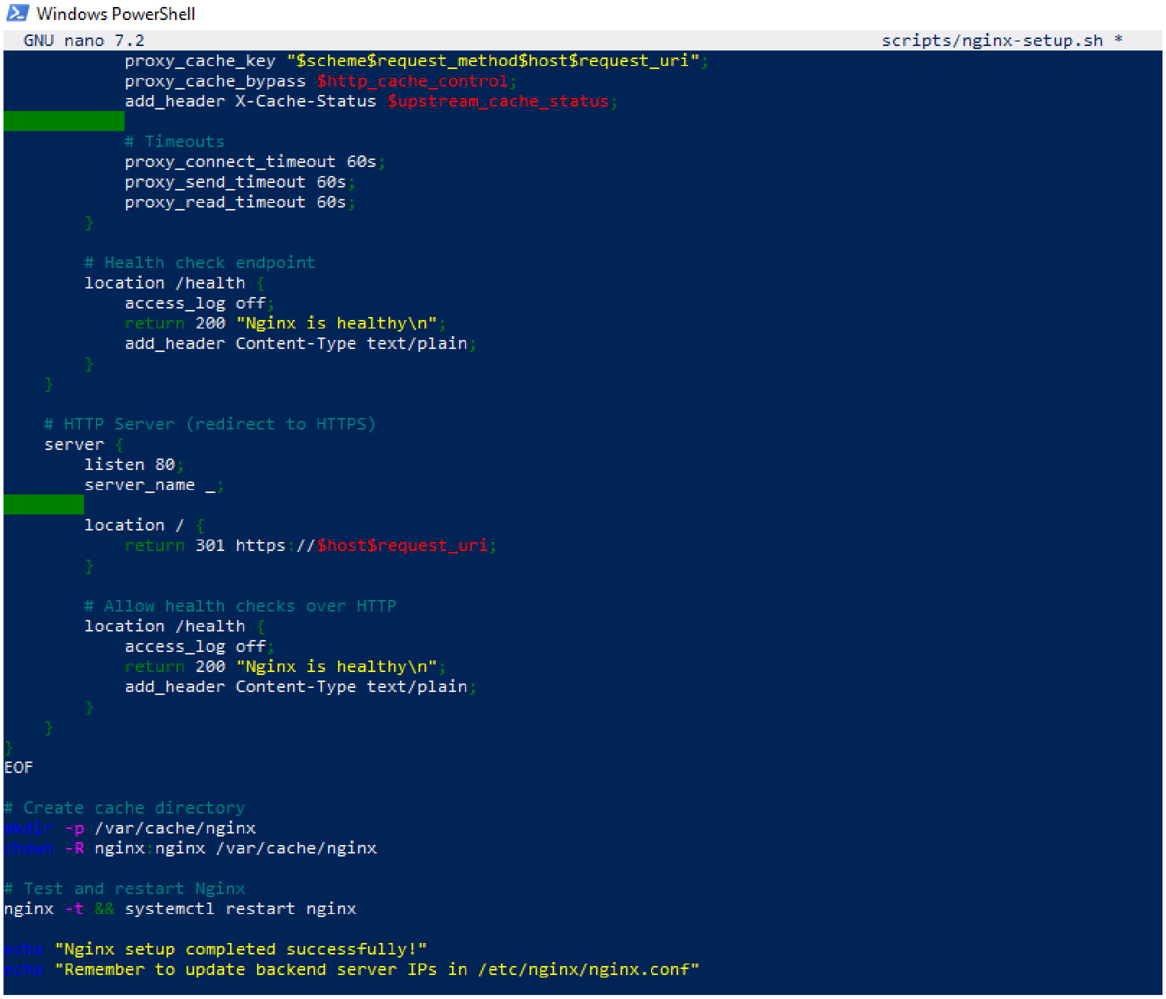
Public DNS:

Deployment Time: Tue Dec 30 09:15:24 UTC 2025

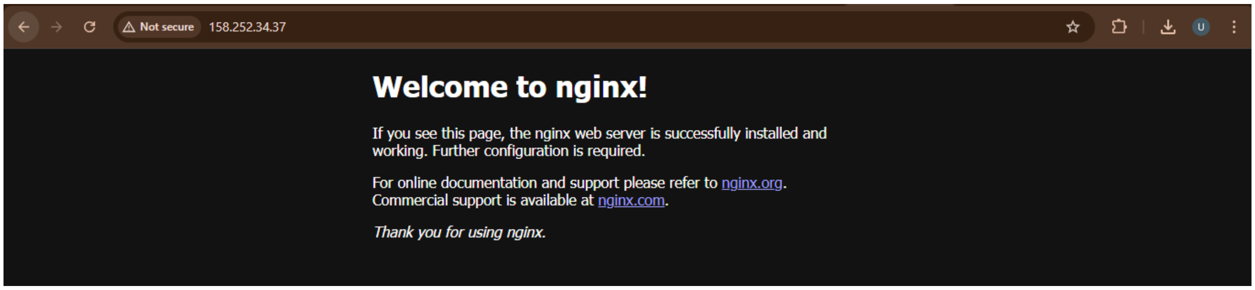
Server Status: Active

Managed By: Terraform

3.2 Nginx Server Setup Script



```
  GNU nano 7.2                                                      scripts/nginx-setup.sh *
            proxy_cache_key "$scheme$request_method$host$request_uri";
            proxy_cache_bypass $http_cache_control;
            add_header X-Cache-Status $upstream_cache_status;

            # Timeouts
            proxy_connect_timeout 60s;
            proxy_send_timeout 60s;
            proxy_read_timeout 60s;
        }

        # Health check endpoint
        location /health {
            access_log off;
            return 200 "Nginx is healthy\n";
            add_header Content-Type text/plain;
        }
    }

    # HTTP Server (redirect to HTTPS)
    server {
        listen 80;
        server_name _;

        location / {
            return 301 https://$host$request_uri;
        }

        # Allow health checks over HTTP
        location /health {
            access_log off;
            return 200 "Nginx is healthy\n";
            add_header Content-Type text/plain;
        }
    }
}
EOF

# Create cache directory
mkdir -p /var/cache/nginx
chown -R nginx:nginx /var/cache/nginx

# Test and restart Nginx
nginx -t && systemctl restart nginx

echo "Nginx setup completed successfully!"
echo "Remember to update backend server IPs in /etc/nginx/nginx.conf"
```



Not secure 158.252.34.37

Welcome to nginx!

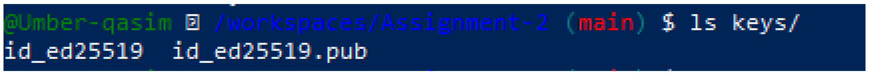
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

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

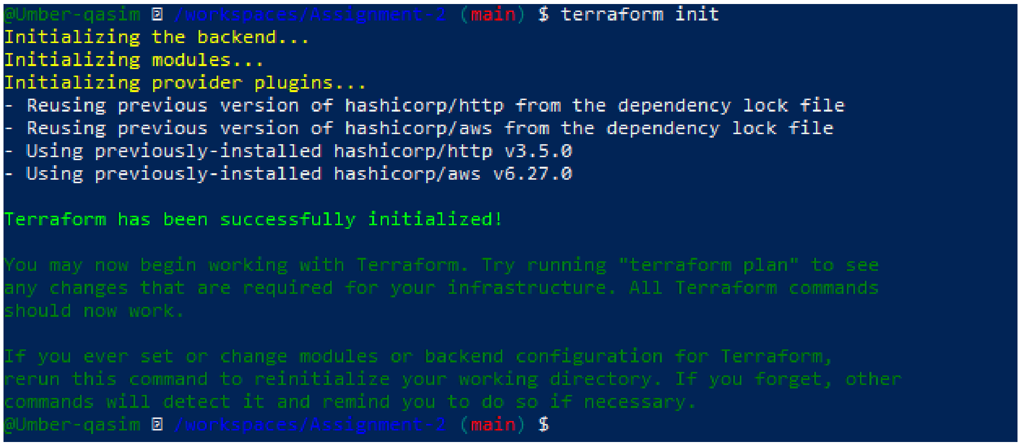
Thank you for using nginx.

Part 4: Infrastructure Deployment

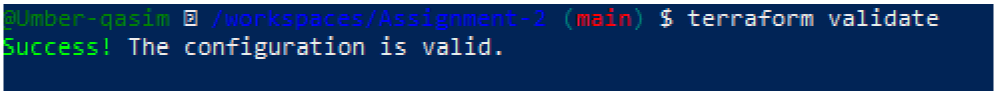
4.1 Initial Deployment



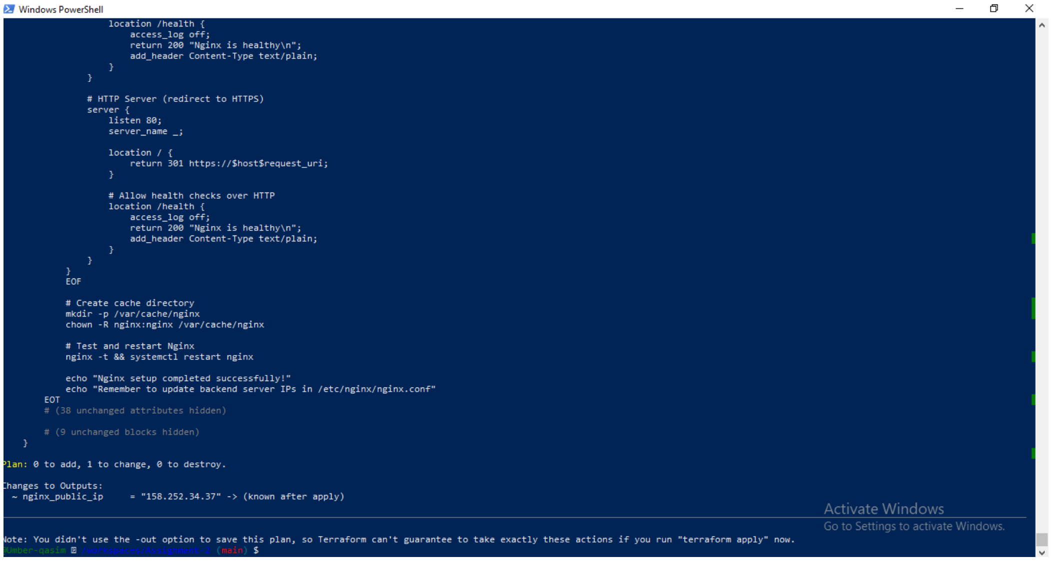
```
@Umber-qasim □ /workspaces/Assignment-2 (main) $ ls keys/
id_ed25519  id_ed25519.pub
```



```
@Umber-qasim □ /workspaces/Assignment-2 (main) $ terraform init
Initializing the backend...
Initializing modules...
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.
@Umber-qasim □ /workspaces/Assignment-2 (main) $
```

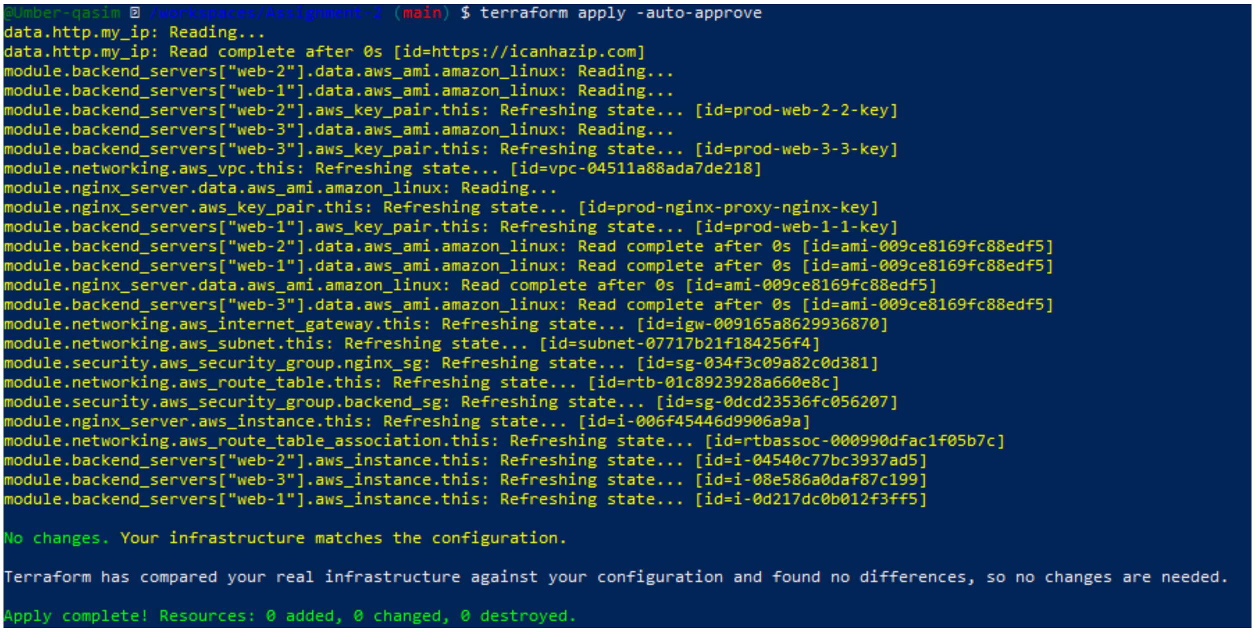


```
@Umber-qasim □ /workspaces/Assignment-2 (main) $ terraform validate
Success! The configuration is valid.
```



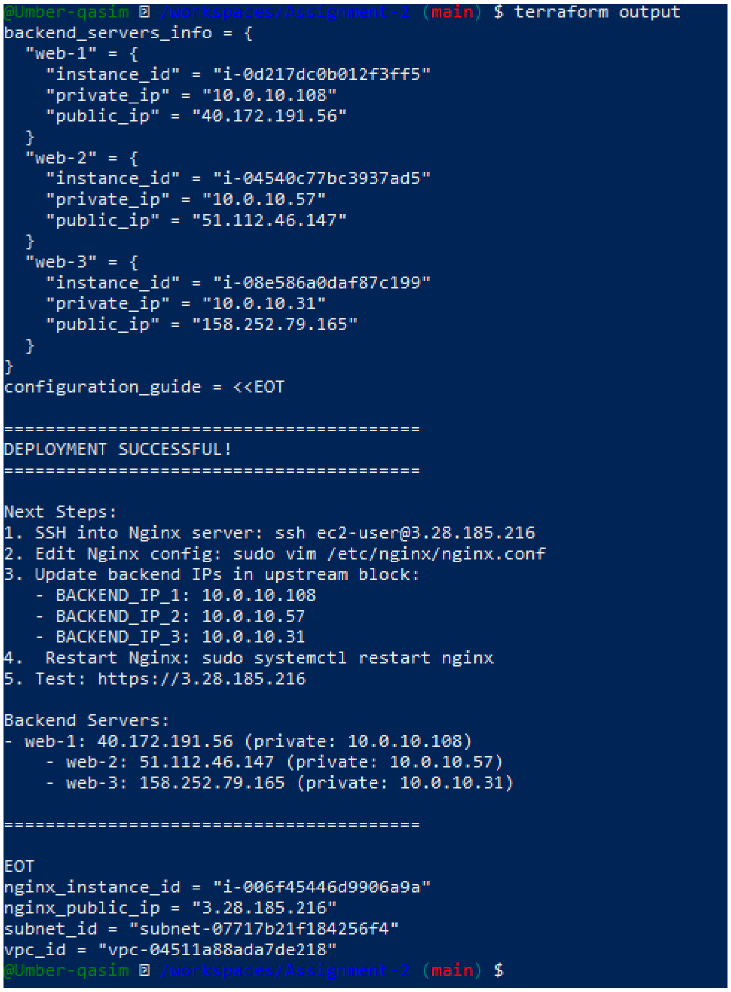
Windows PowerShell

```
                    location /health {
                        access_log off;
                        return 200 "Nginx is healthy\n";
                        add_header Content-Type text/plain;
                    }
                }

                # HTTP Server (redirect to HTTPS)
                server {
                    listen 80;
                    server_name _;

                    location / {
                        return 301 https://$host$request_uri;
                    }

                    # Allow health checks over HTTP
                    location /health {
                        access_log off;
                        return 200 "Nginx is healthy\n";
                        add_header Content-Type text/plain;
                    }
                }
            }
            EOF

            # Create cache directory
            mkdir -p /var/cache/nginx
            chown -R nginx:nginx /var/cache/nginx

            # Test and restart Nginx
            nginx -t && systemctl restart nginx

            echo "Nginx setup completed successfully!"
            echo "Remember to update backend server IPs in /etc/nginx/nginx.conf"
        EOT
        # (38 unchanged attributes hidden)

        # (9 unchanged blocks hidden)
    }

Plan: 0 to add, 1 to change, 0 to destroy.

Changes to Outputs:
  ~ nginx_public_ip     = "158.252.34.37" -> (known after apply)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
@Umber-qasim □ /workspaces/Assignment-2 (main) $
```

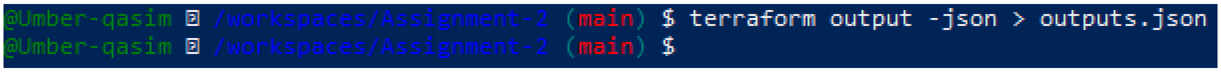


```
@Umber-qasim ⊠ /workspaces/Assignment-2 (main) $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
module.backend_servers["web-2"].data.aws_ami.amazon_linux: Reading...
module.backend_servers["web-1"].data.aws_ami.amazon_linux: Reading...
module.backend_servers["web-2"].aws_key_pair.this: Refreshing state... [id=prod-web-2-2-key]
module.backend_servers["web-3"].data.aws_ami.amazon_linux: Reading...
module.backend_servers["web-3"].aws_key_pair.this: Refreshing state... [id=prod-web-3-3-key]
module.networking.aws_vpc.this: Refreshing state... [id=vpc-04511a88ada7de218]
module.nginx_server.data.aws_ami.amazon_linux: Reading...
module.nginx_server.aws_key_pair.this: Refreshing state... [id=prod-nginx-proxy-nginx-key]
module.backend_servers["web-1"].aws_key_pair.this: Refreshing state... [id=prod-web-1-1-key]
module.backend_servers["web-2"].data.aws_ami.amazon_linux: Read complete after 0s [id=ami-009ce8169fc88edf5]
module.backend_servers["web-1"].data.aws_ami.amazon_linux: Read complete after 0s [id=ami-009ce8169fc88edf5]
module.nginx_server.data.aws_ami.amazon_linux: Read complete after 0s [id=ami-009ce8169fc88edf5]
module.backend_servers["web-3"].data.aws_ami.amazon_linux: Read complete after 0s [id=ami-009ce8169fc88edf5]
module.networking.aws_internet_gateway.this: Refreshing state... [id=igw-009165a8629936870]
module.networking.aws_subnet.this: Refreshing state... [id=subnet-07717b21f184256f4]
module.security.aws_security_group.nginx_sg: Refreshing state... [id=sg-034f3c09a82c0d381]
module.networking.aws_route_table.this: Refreshing state... [id=rtb-01c8923928a660e8c]
module.security.aws_security_group.backend_sg: Refreshing state... [id=sg-0dcd23536fc056207]
module.nginx_server.aws_instance.this: Refreshing state... [id=i-006f45446d9906a9a]
module.networking.aws_route_table_association.this: Refreshing state... [id=rtbassoc-000990dfac1f05b7c]
module.backend_servers["web-2"].aws_instance.this: Refreshing state... [id=i-04540c77bc3937ad5]
module.backend_servers["web-3"].aws_instance.this: Refreshing state... [id=i-08e586a0daf87c199]
module.backend_servers["web-1"].aws_instance.this: Refreshing state... [id=i-0d217dc0b012f3ff5]

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

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

4.2 Output Configuration

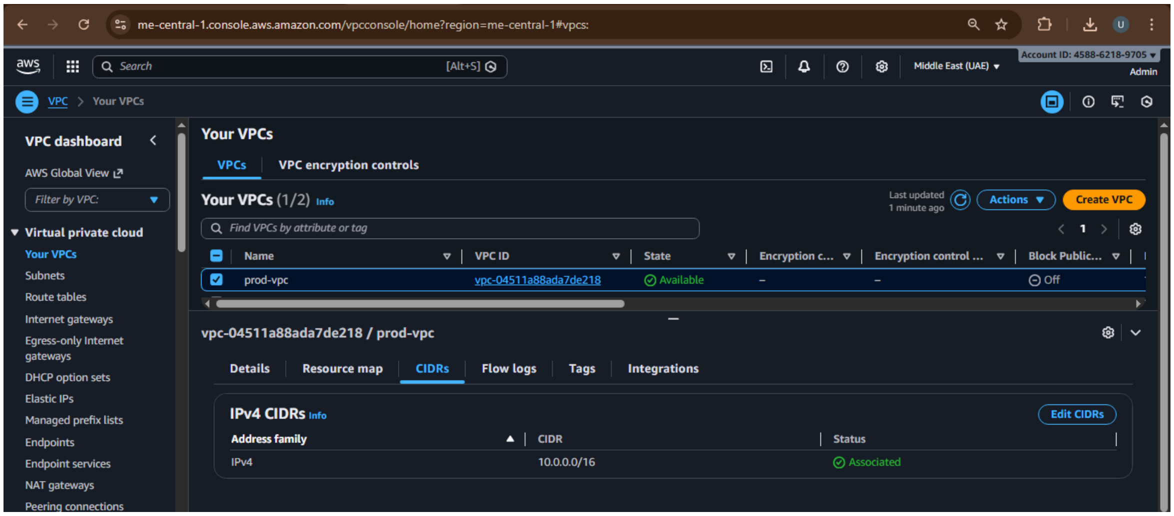


```
@Umber-qasim ⊠ /workspaces/Assignment-2 (main) $ terraform output
backend_servers_info = {
  "web-1" = {
    "instance_id" = "i-0d217dc0b012f3ff5"
    "private_ip" = "10.0.10.108"
    "public_ip" = "40.172.191.56"
  }
  "web-2" = {
    "instance_id" = "i-04540c77bc3937ad5"
    "private_ip" = "10.0.10.57"
    "public_ip" = "51.112.46.147"
  }
  "web-3" = {
    "instance_id" = "i-08e586a0daf87c199"
    "private_ip" = "10.0.10.31"
    "public_ip" = "158.252.79.165"
  }
}
configuration_guide = <<EOT

========================================
DEPLOYMENT SUCCESSFUL!
========================================

Next Steps:
1. SSH into Nginx server: ssh ec2-user@3.28.185.216
2. Edit Nginx config: sudo vim /etc/nginx/nginx.conf
3. Update backend IPs in upstream block:
   - BACKEND_IP_1: 10.0.10.108
   - BACKEND_IP_2: 10.0.10.57
   - BACKEND_IP_3: 10.0.10.31
4.  Restart Nginx: sudo systemctl restart nginx
5. Test: https://3.28.185.216

Backend Servers:
- web-1: 40.172.191.56 (private: 10.0.10.108)
    - web-2: 51.112.46.147 (private: 10.0.10.57)
    - web-3: 158.252.79.165 (private: 10.0.10.31)

========================================

EOT
nginx_instance_id = "i-006f45446d9906a9a"
nginx_public_ip = "3.28.185.216"
subnet_id = "subnet-07717b21f184256f4"
vpc_id = "vpc-04511a88ada7de218"
@Umber-qasim ⊠ /workspaces/Assignment-2 (main) $
```



```
@Umber-qasim ▯ /workspaces/Assignment-2 (main) $ terraform output -json > outputs.json
@Umber-qasim ▯ /workspaces/Assignment-2 (main) $
```

4.3 AWS Console Verification



me-central-1.console.aws.amazon.com/vpcconsole/home?region=me-central-1#vpcs:

Account ID: 4588-6218-9705

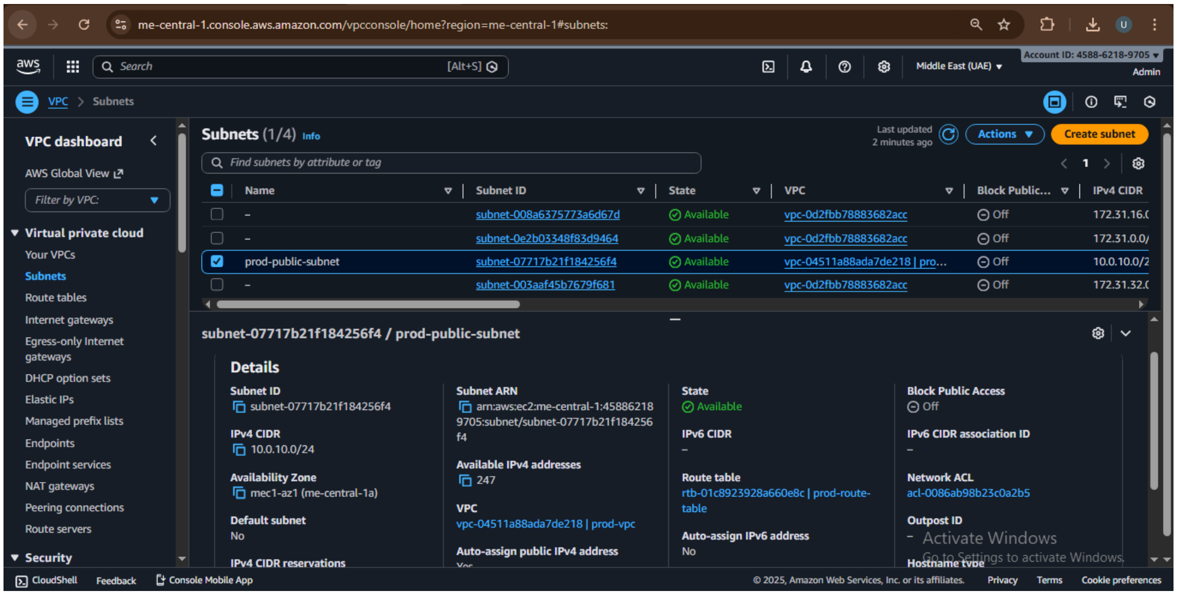
Your VPCs (1/2)

Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...
prod-vpc	vpc-04511a88ada7de218	Available	–	–	Off

vpc-04511a88ada7de218 / prod-vpc

IPv4 CIDRs

Address family	CIDR	Status
IPv4	10.0.0.0/16	Associated



me-central-1.console.aws.amazon.com/vpcconsole/home?region=me-central-1#subnets:

Account ID: 4588-6218-9705

Subnets (1/4)

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR
–	subnet-008a6375773a6d67d	Available	vpc-0d2fbb78883682acc	Off	172.31.16.(
–	subnet-0e2b03348f83d9464	Available	vpc-0d2fbb78883682acc	Off	172.31.0.0/
prod-public-subnet	subnet-07717b21f184256f4	Available	vpc-04511a88ada7de218 | pro...	Off	10.0.10.0/2
–	subnet-003aaf45b7679f681	Available	vpc-0d2fbb78883682acc	Off	172.31.32.(

subnet-07717b21f184256f4 / prod-public-subnet

Details

Subnet ID
subnet-07717b21f184256f4

IPv4 CIDR
10.0.10.0/24

Availability Zone
mec1-az1 (me-central-1a)

Default subnet
No

IPv4 CIDR reservations

Subnet ARN
arn:aws:ec2:me-central-1:458862189705:subnet/subnet-07717b21f184256f4

Available IPv4 addresses
247

VPC
vpc-04511a88ada7de218 | prod-vpc

Auto-assign public IPv4 address

State
Available

IPv6 CIDR
–

Route table
rtb-01c8923928a660e8c | prod-route-table

Auto-assign IPv6 address
No

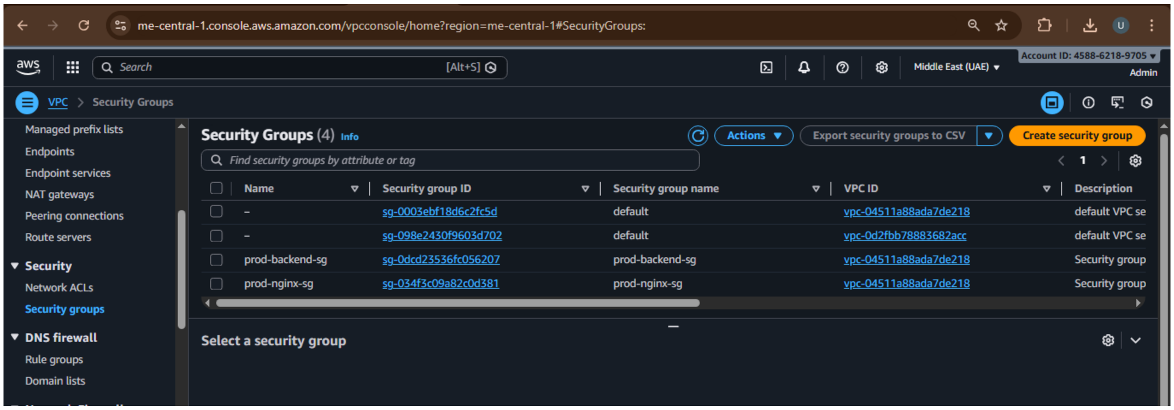
Block Public Access
Off

IPv6 CIDR association ID
–

Network ACL
acl-0086ab98b23c0a2b5

Outpost ID
–

Hostname type



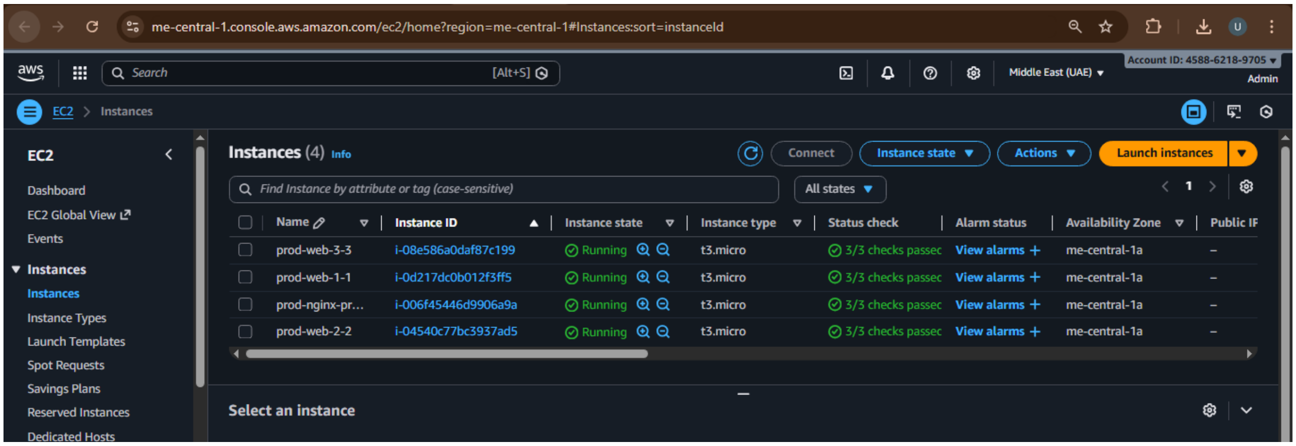
me-central-1.console.aws.amazon.com/vpcconsole/home?region=me-central-1#SecurityGroups:

Account ID: 4588-6218-9705

Security Groups (4)

Name	Security group ID	Security group name	VPC ID	Description
–	sg-0003ebf18d6c2fc5d	default	vpc-04511a88ada7de218	default VPC se
–	sg-098e2430f9603d702	default	vpc-0d2fbb78883682acc	default VPC se
prod-backend-sg	sg-0dcd23536fc056207	prod-backend-sg	vpc-04511a88ada7de218	Security group
prod-nginx-sg	sg-034f3c09a82c0d381	prod-nginx-sg	vpc-04511a88ada7de218	Security group

Select a security group

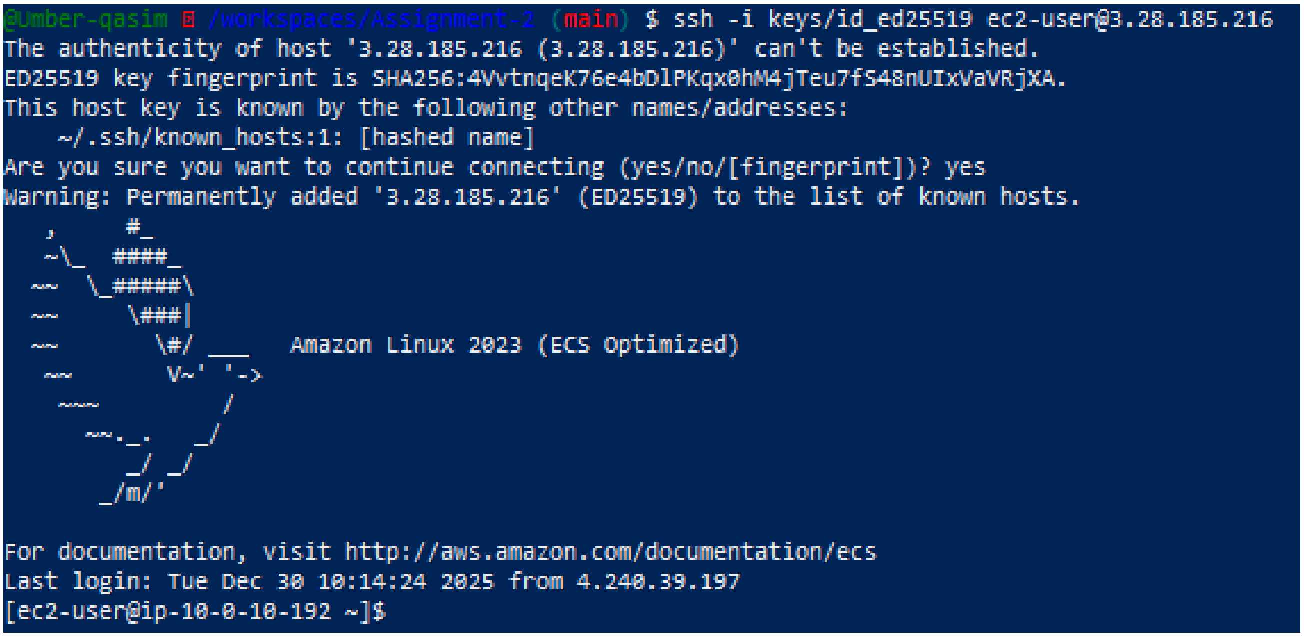


Instances (4) Info

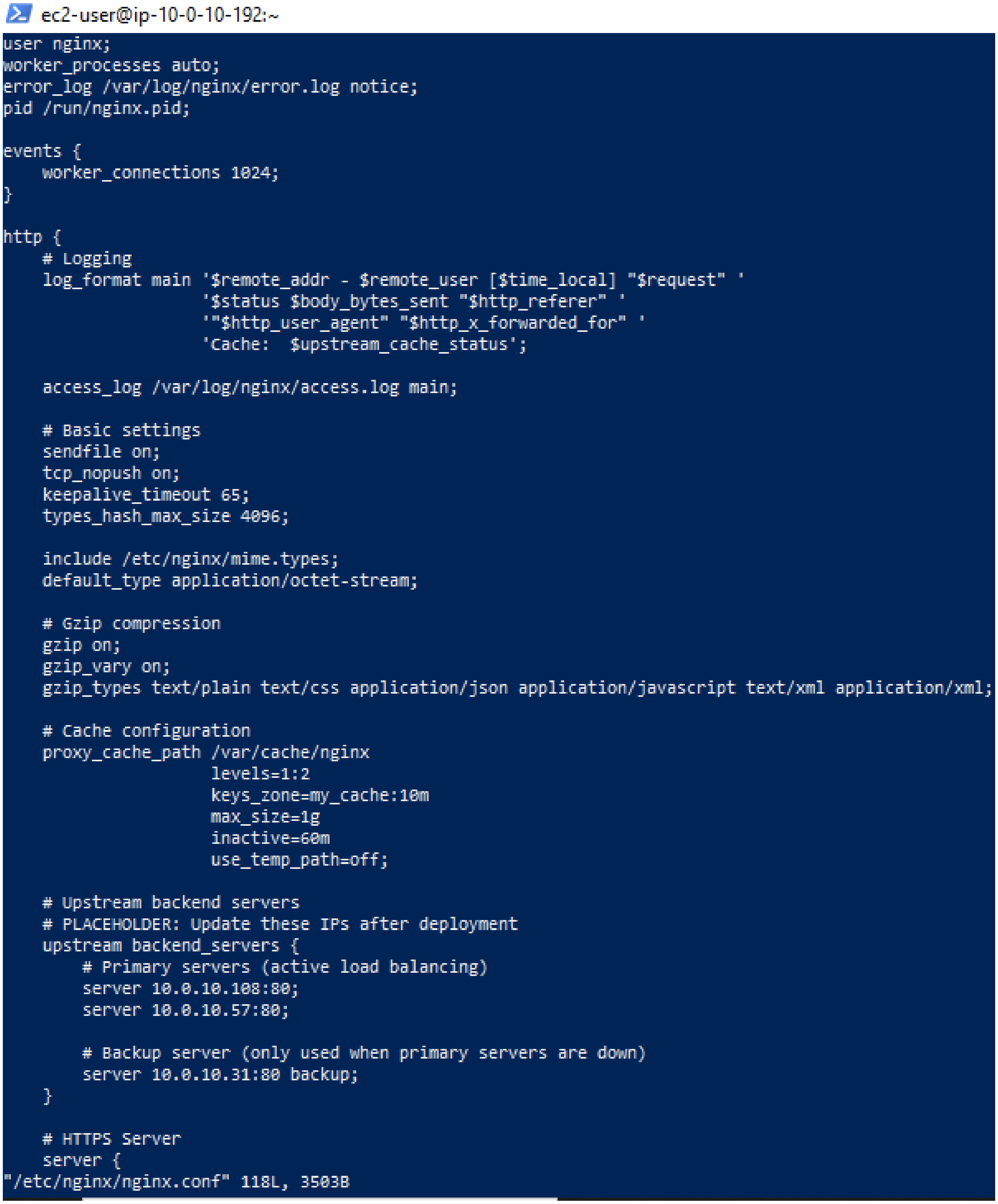
☐	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
☐	prod-web-3-3	i-08e586a0daf87c199	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	me-central-1a	–
☐	prod-web-1-1	i-0d217dc0b012f3ff5	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	me-central-1a	–
☐	prod-nginx-pr...	i-006f45446d9906a9a	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	me-central-1a	–
☐	prod-web-2-2	i-04540c77bc3937ad5	✓ Running	t3.micro	✓ 3/3 checks passec	View alarms +	me-central-1a	–

Part 5: Nginx Configuration & Testing

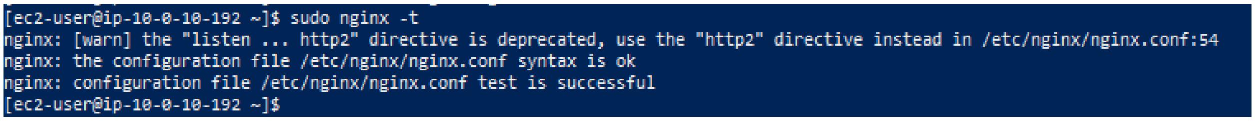
5.1 Update Nginx Backend Configuration



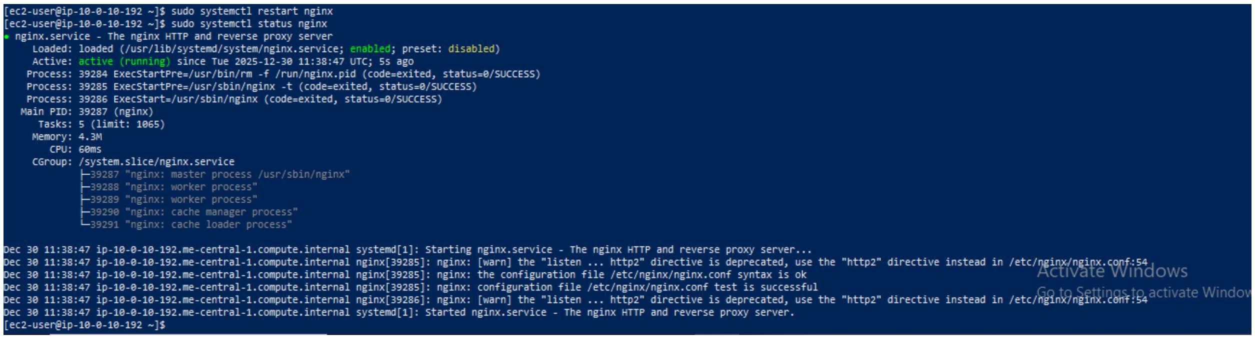
```
@Umber-qasim ⊠ /workspaces/Assignment-2 (main) $ ssh -i keys/id_ed25519 ec2-user@3.28.185.216
The authenticity of host '3.28.185.216 (3.28.185.216)' can't be established.
ED25519 key fingerprint is SHA256:4VvtnqeK76e4bDlPKqx0hM4jTeu7fS48nUIxVaVRjXA.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.28.185.216' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   Amazon Linux 2023 (ECS Optimized)
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

For documentation, visit http://aws.amazon.com/documentation/ecs
Last login: Tue Dec 30 10:14:24 2025 from 4.240.39.197
[ec2-user@ip-10-0-10-192 ~]$
```



```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log notice;
pid /run/nginx.pid;

events {
    worker_connections 1024;
}

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

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

    # Basic settings
    sendfile on;
    tcp_nopush on;
    keepalive_timeout 65;
    types_hash_max_size 4096;

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

    # Gzip compression
    gzip on;
    gzip_vary on;
    gzip_types text/plain text/css application/json application/javascript text/xml application/xml;

    # Cache configuration
    proxy_cache_path /var/cache/nginx
                     levels=1:2
                     keys_zone=my_cache:10m
                     max_size=1g
                     inactive=60m
                     use_temp_path=off;

    # Upstream backend servers
    # PLACEHOLDER: Update these IPs after deployment
    upstream backend_servers {
        # Primary servers (active load balancing)
        server 10.0.10.108:80;
        server 10.0.10.57:80;

        # Backup server (only used when primary servers are down)
        server 10.0.10.31:80 backup;
    }

    # HTTPS Server
    server {
"/etc/nginx/nginx.conf" 118L, 3503B
```

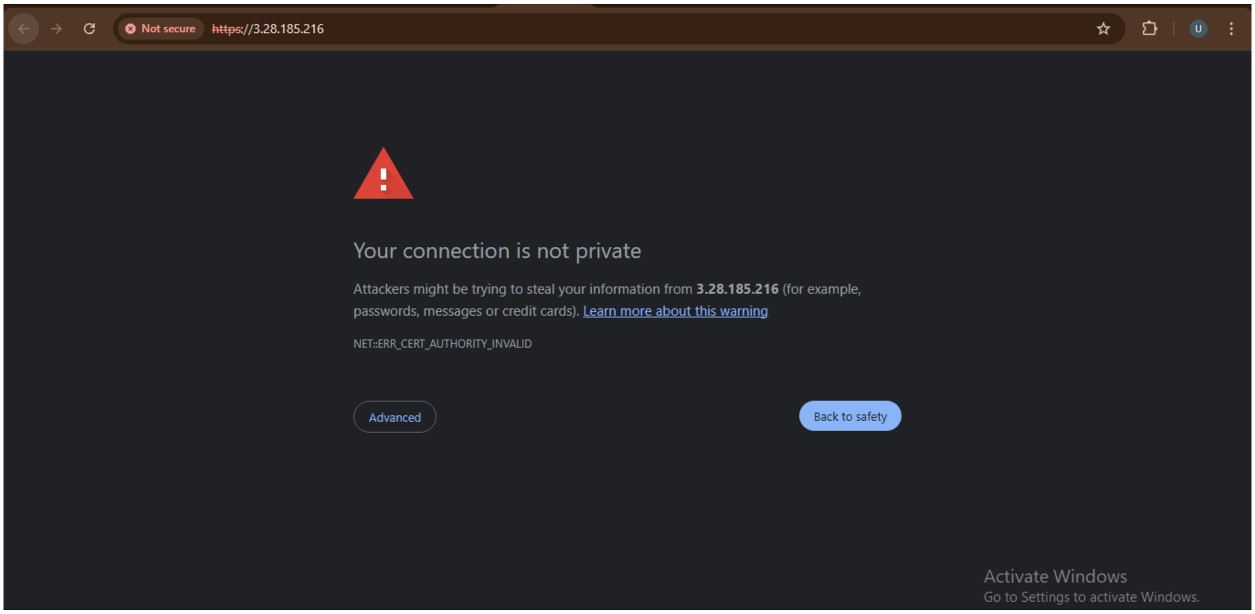


```
[ec2-user@ip-10-0-10-192 ~]$ sudo nginx -t
nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
[ec2-user@ip-10-0-10-192 ~]$
```



```
[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-30 11:38:47 UTC; 5s ago
    Process: 39284 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 39285 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 39286 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 39287 (nginx)
      Tasks: 5 (limit: 1065)
     Memory: 4.3M
        CPU: 60ms
     CGroup: /system.slice/nginx.service
             ├─39287 "nginx: master process /usr/sbin/nginx"
             ├─39288 "nginx: worker process"
             ├─39289 "nginx: worker process"
             ├─39290 "nginx: cache manager process"
             └─39291 "nginx: cache loader process"

Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal nginx[39285]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal nginx[39285]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal nginx[39285]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal nginx[39286]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
Dec 30 11:38:47 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
[ec2-user@ip-10-0-10-192 ~]$
```

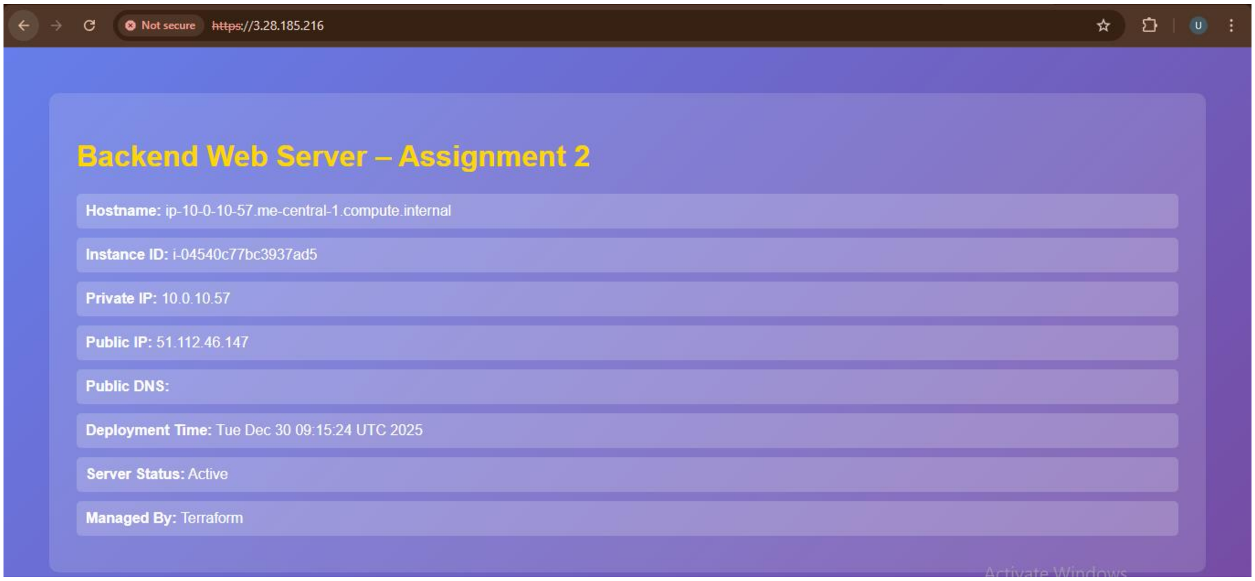
5.2 Test Load Balancing



Your connection is not private

Attackers might be trying to steal your information from **3.28.185.216** (for example, passwords, messages or credit cards). Learn more about this warning

NET::ERR_CERT_AUTHORITY_INVALID



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-57.me-central-1.compute.internal

Instance ID: i-04540c77bc3937ad5

Private IP: 10.0.10.57

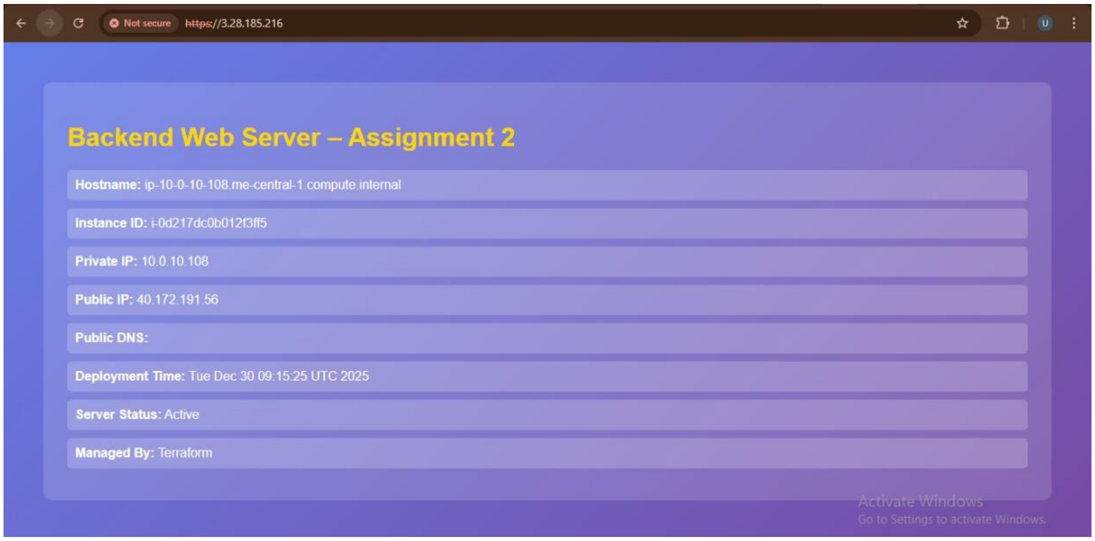
Public IP: 51.112.46.147

Public DNS:

Deployment Time: Tue Dec 30 09:15:24 UTC 2025

Server Status: Active

Managed By: Terraform



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-108.me-central-1.compute.internal

Instance ID: i-0d217dc0b012f3ff5

Private IP: 10.0.10.108

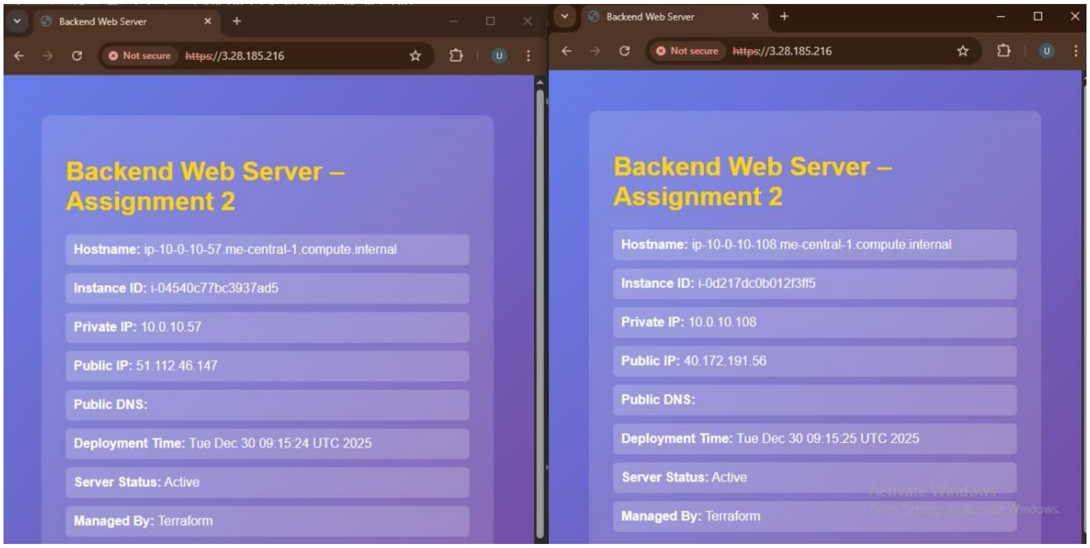
Public IP: 40.172.191.56

Public DNS:

Deployment Time: Tue Dec 30 09:15:25 UTC 2025

Server Status: Active

Managed By: Terraform



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-57.me-central-1.compute.internal

Instance ID: i-04540c77bc3937ad5

Private IP: 10.0.10.57

Public IP: 51.112.46.147

Public DNS:

Deployment Time: Tue Dec 30 09:15:24 UTC 2025

Server Status: Active

Managed By: Terraform

Backend Web Server – Assignment 2

Hostname: ip-10-0-10-108.me-central-1.compute.internal

Instance ID: i-0d217dc0b012f3ff5

Private IP: 10.0.10.108

Public IP: 40.172.191.56

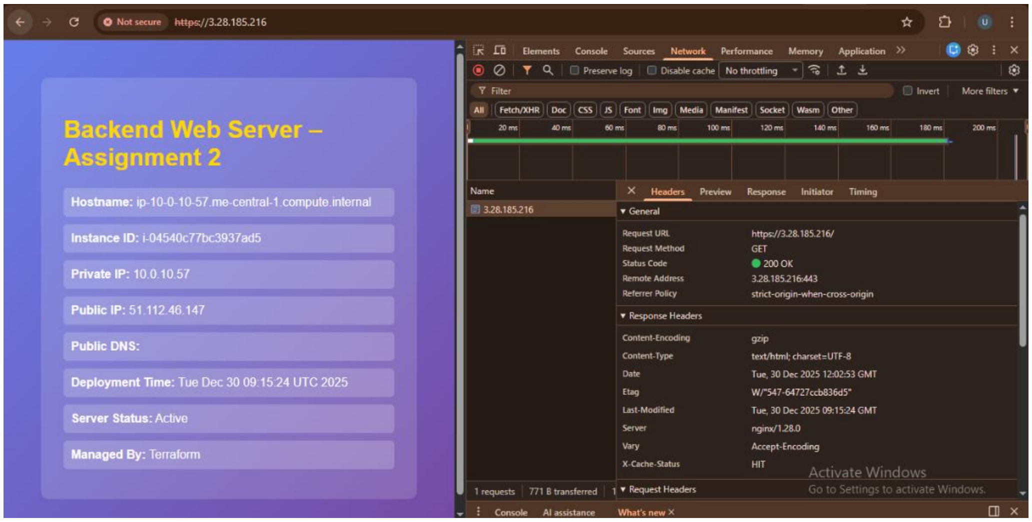
Public DNS:

Deployment Time: Tue Dec 30 09:15:25 UTC 2025

Server Status: Active

Managed By: Terraform

5.3 Test Cache Functionality



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-57.me-central-1.compute.internal

Instance ID: i-04540c77bc3937ad5

Private IP: 10.0.10.57

Public IP: 51.112.46.147

Public DNS:

Deployment Time: Tue Dec 30 09:15:24 UTC 2025

Server Status: Active

Managed By: Terraform

3.28.185.216

▾ General

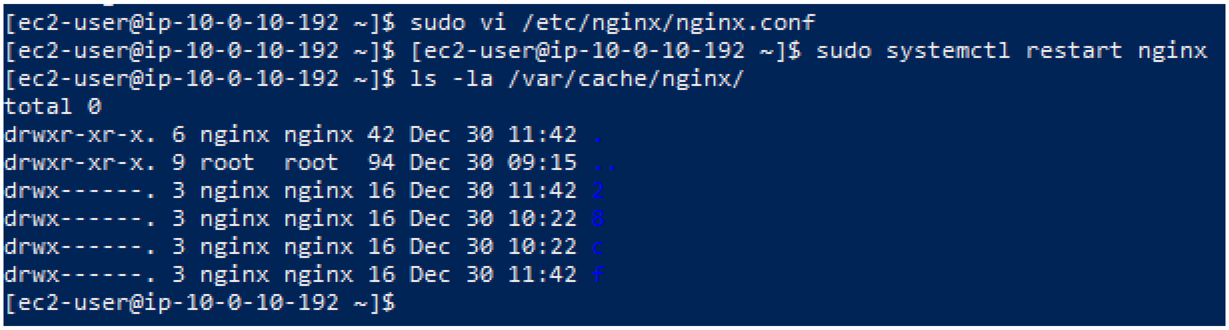
Request URL	https://3.28.185.216/
Request Method	GET
Status Code	● 200 OK
Remote Address	3.28.185.216:443
Referrer Policy	strict-origin-when-cross-origin

▾ Response Headers

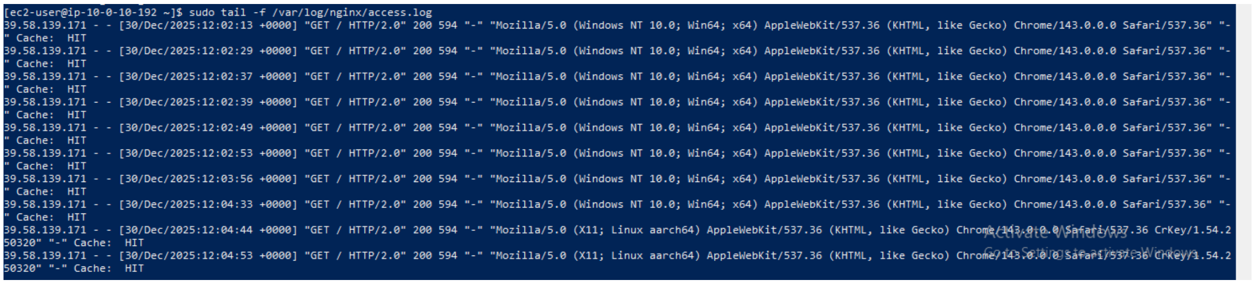
Content-Encoding	gzip
Content-Type	text/html; charset=UTF-8
Date	Tue, 30 Dec 2025 12:02:53 GMT
Etag	W/"547-64727ccb836d5"
Last-Modified	Tue, 30 Dec 2025 09:15:24 GMT
Server	nginx/1.28.0
Vary	Accept-Encoding
X-Cache-Status	HIT

▾ Request Headers

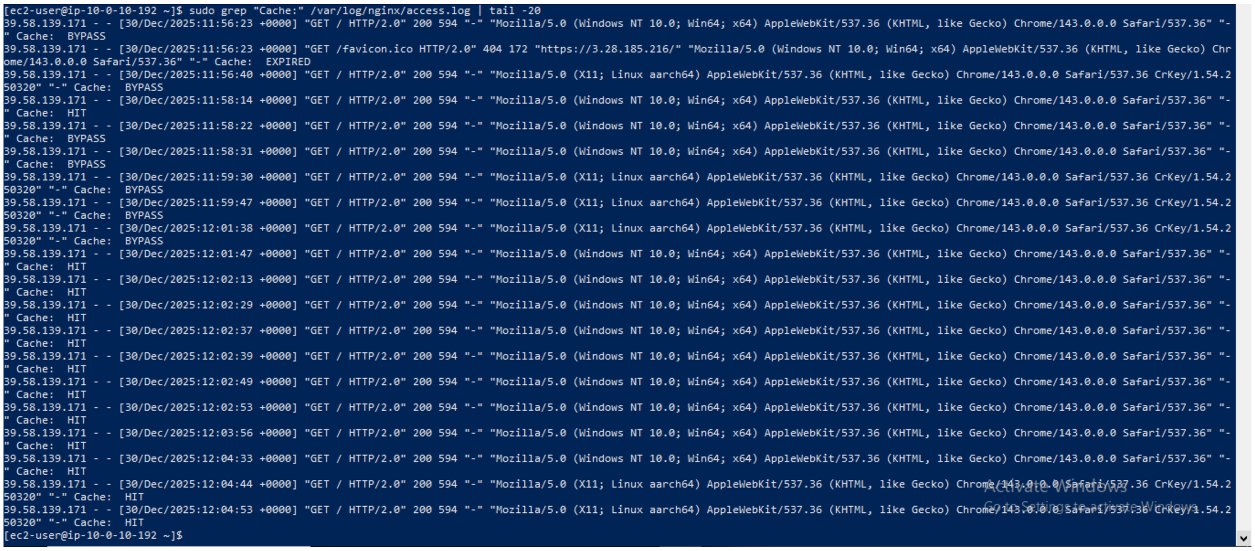
1 requests | 771 B transferred



```
[ec2-user@ip-10-0-10-192 ~]$ sudo vi /etc/nginx/nginx.conf
[ec2-user@ip-10-0-10-192 ~]$ [ec2-user@ip-10-0-10-192 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-192 ~]$ ls -la /var/cache/nginx/
total 0
drwxr-xr-x. 6 nginx nginx 42 Dec 30 11:42 .
drwxr-xr-x. 9 root  root  94 Dec 30 09:15 ..
drwx------. 3 nginx nginx 16 Dec 30 11:42 2
drwx------. 3 nginx nginx 16 Dec 30 10:22 8
drwx------. 3 nginx nginx 16 Dec 30 10:22 c
drwx------. 3 nginx nginx 16 Dec 30 11:42 f
[ec2-user@ip-10-0-10-192 ~]$
```

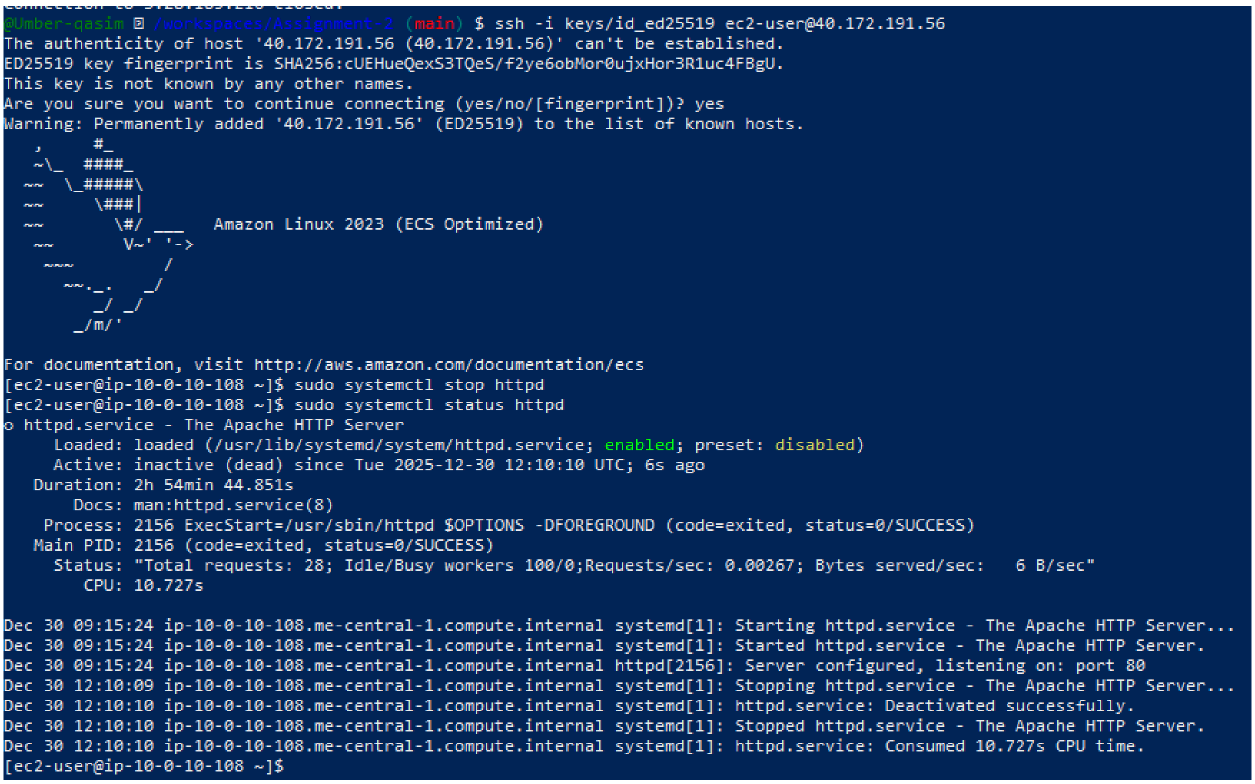


```
[ec2-user@ip-10-0-10-192 ~]$ sudo tail -f /var/log/nginx/access.log
39.58.139.171 - - [30/Dec/2025:12:02:13 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:29 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:37 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:39 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:49 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:53 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:03:56 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:33 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:44 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:53 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  HIT
```

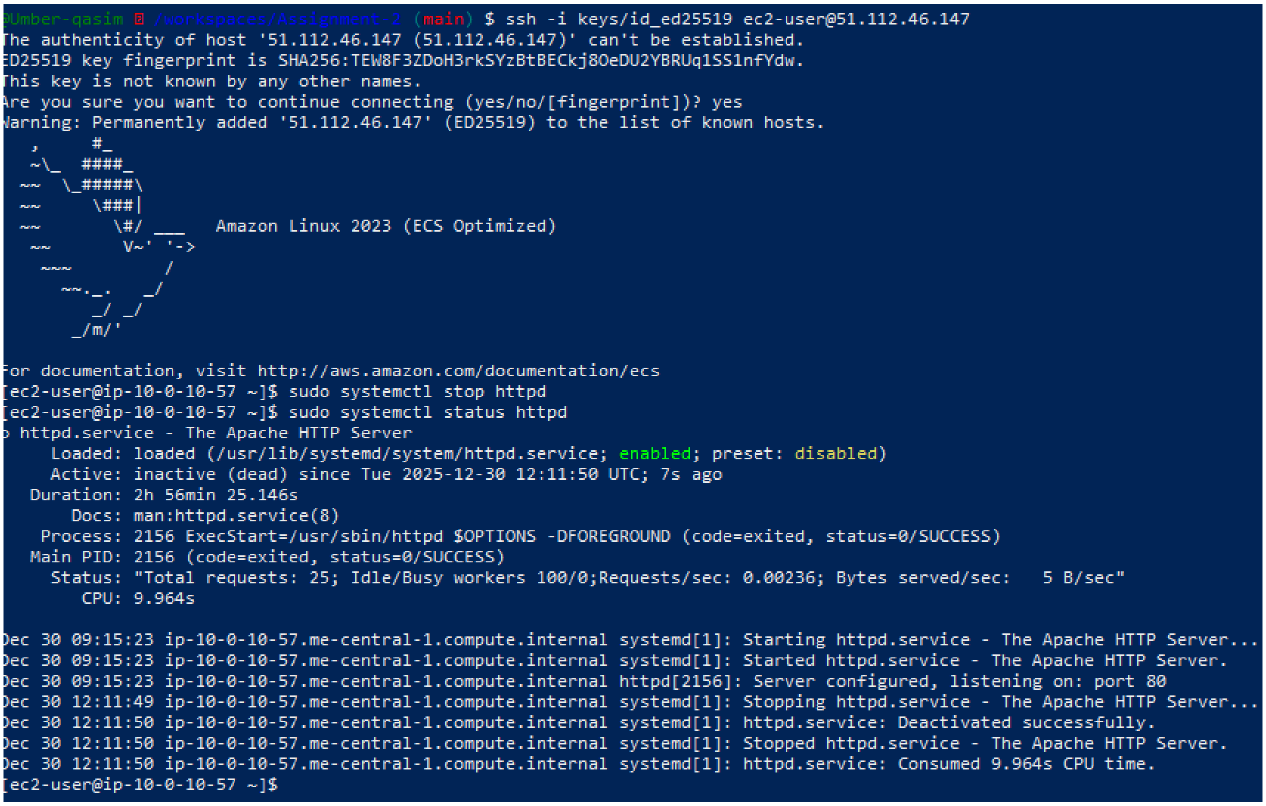


```
[ec2-user@ip-10-0-10-192 ~]$ sudo grep "Cache:" /var/log/nginx/access.log | tail -20
39.58.139.171 - - [30/Dec/2025:11:56:23 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:11:56:23 +0000] "GET /favicon.ico HTTP/2.0" 404 172 "https://3.28.185.216/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chr
ome/143.0.0.0 Safari/537.36" "-" Cache:  EXPIRED
39.58.139.171 - - [30/Dec/2025:11:56:40 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:11:58:14 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:11:58:22 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:11:58:31 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:11:59:30 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:11:59:47 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:12:01:38 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  BYPASS
39.58.139.171 - - [30/Dec/2025:12:01:47 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:13 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:29 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:37 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:39 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:49 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:02:53 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:03:56 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:33 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:44 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:12:04:53 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (X11; Linux aarch64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 CrKey/1.54.2
50320" "-" Cache:  HIT
[ec2-user@ip-10-0-10-192 ~]$
```

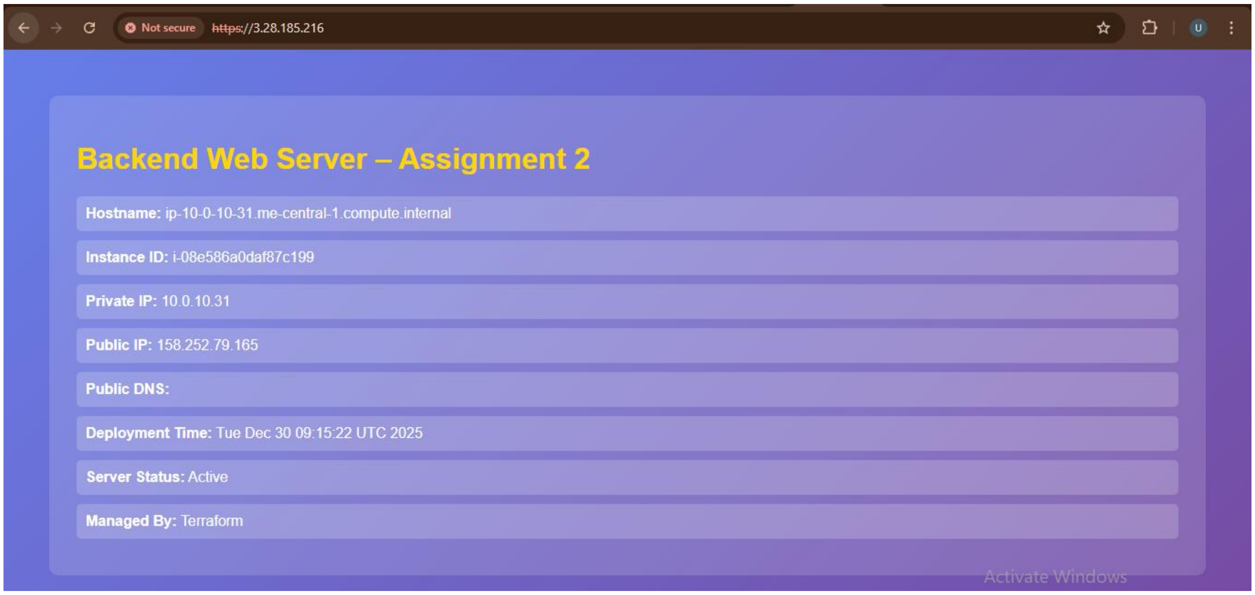
5.4 Test High Availability (Backup Server)



```
@Umber-qasim ⍰ /workspaces/Assignment-2 (main) $ ssh -i keys/id_ed25519 ec2-user@40.172.191.56
The authenticity of host '40.172.191.56 (40.172.191.56)' can't be established.
ED25519 key fingerprint is SHA256:cUEHueQexS3TQeS/f2ye6obMor0ujxHor3R1uc4FBgU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '40.172.191.56' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   Amazon Linux 2023 (ECS Optimized)
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

For documentation, visit http://aws.amazon.com/documentation/ecs
[ec2-user@ip-10-0-10-108 ~]$ sudo systemctl stop httpd
[ec2-user@ip-10-0-10-108 ~]$ sudo systemctl status httpd
○ httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
     Active: inactive (dead) since Tue 2025-12-30 12:10:10 UTC; 6s ago
   Duration: 2h 54min 44.851s
       Docs: man:httpd.service(8)
    Process: 2156 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=exited, status=0/SUCCESS)
   Main PID: 2156 (code=exited, status=0/SUCCESS)
     Status: "Total requests: 28; Idle/Busy workers 100/0;Requests/sec: 0.00267; Bytes served/sec:   6 B/sec"
        CPU: 10.727s

Dec 30 09:15:24 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Dec 30 09:15:24 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Dec 30 09:15:24 ip-10-0-10-108.me-central-1.compute.internal httpd[2156]: Server configured, listening on: port 80
Dec 30 12:10:09 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: Stopping httpd.service - The Apache HTTP Server...
Dec 30 12:10:10 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: httpd.service: Deactivated successfully.
Dec 30 12:10:10 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: Stopped httpd.service - The Apache HTTP Server.
Dec 30 12:10:10 ip-10-0-10-108.me-central-1.compute.internal systemd[1]: httpd.service: Consumed 10.727s CPU time.
[ec2-user@ip-10-0-10-108 ~]$
```



```
@Umber-qasim ⍰ /workspaces/Assignment-2 (main) $ ssh -i keys/id_ed25519 ec2-user@51.112.46.147
The authenticity of host '51.112.46.147 (51.112.46.147)' can't be established.
ED25519 key fingerprint is SHA256:TEW8F3ZDoH3rkSYzBtBECkj8OeDU2YBRUq1SS1nfYdw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '51.112.46.147' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   Amazon Linux 2023 (ECS Optimized)
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

For documentation, visit http://aws.amazon.com/documentation/ecs
[ec2-user@ip-10-0-10-57 ~]$ sudo systemctl stop httpd
[ec2-user@ip-10-0-10-57 ~]$ sudo systemctl status httpd
○ httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
     Active: inactive (dead) since Tue 2025-12-30 12:11:50 UTC; 7s ago
   Duration: 2h 56min 25.146s
       Docs: man:httpd.service(8)
    Process: 2156 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=exited, status=0/SUCCESS)
   Main PID: 2156 (code=exited, status=0/SUCCESS)
     Status: "Total requests: 25; Idle/Busy workers 100/0;Requests/sec: 0.00236; Bytes served/sec:   5 B/sec"
        CPU: 9.964s

Dec 30 09:15:23 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Dec 30 09:15:23 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Dec 30 09:15:23 ip-10-0-10-57.me-central-1.compute.internal httpd[2156]: Server configured, listening on: port 80
Dec 30 12:11:49 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: Stopping httpd.service - The Apache HTTP Server...
Dec 30 12:11:50 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: httpd.service: Deactivated successfully.
Dec 30 12:11:50 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: Stopped httpd.service - The Apache HTTP Server.
Dec 30 12:11:50 ip-10-0-10-57.me-central-1.compute.internal systemd[1]: httpd.service: Consumed 9.964s CPU time.
[ec2-user@ip-10-0-10-57 ~]$
```



Backend Web Server – Assignment 2

Hostname: ip-10-0-10-31.me-central-1.compute.internal

Instance ID: i-08e586a0daf87c199

Private IP: 10.0.10.31

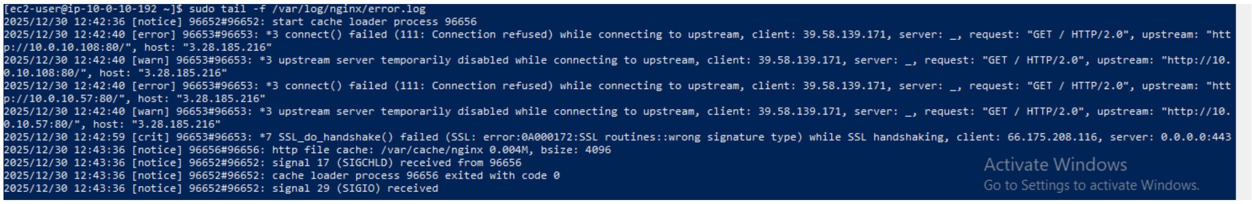
Public IP: 158.252.79.165

Public DNS:

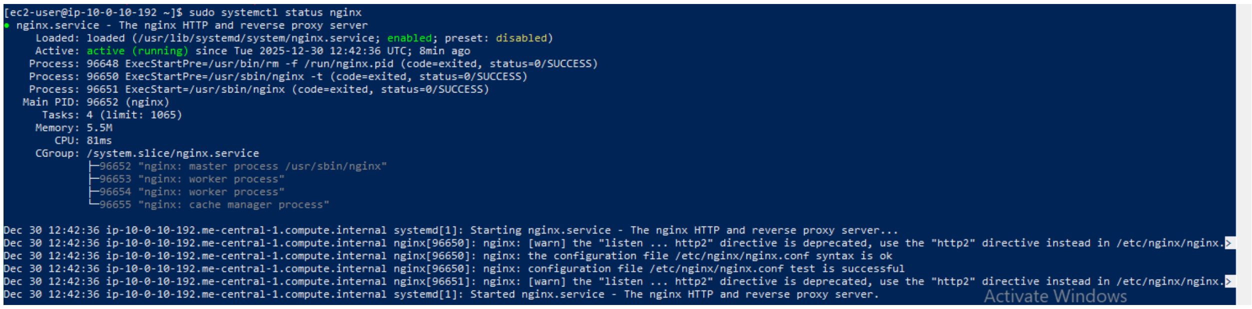
Deployment Time: Tue Dec 30 09:15:22 UTC 2025

Server Status: Active

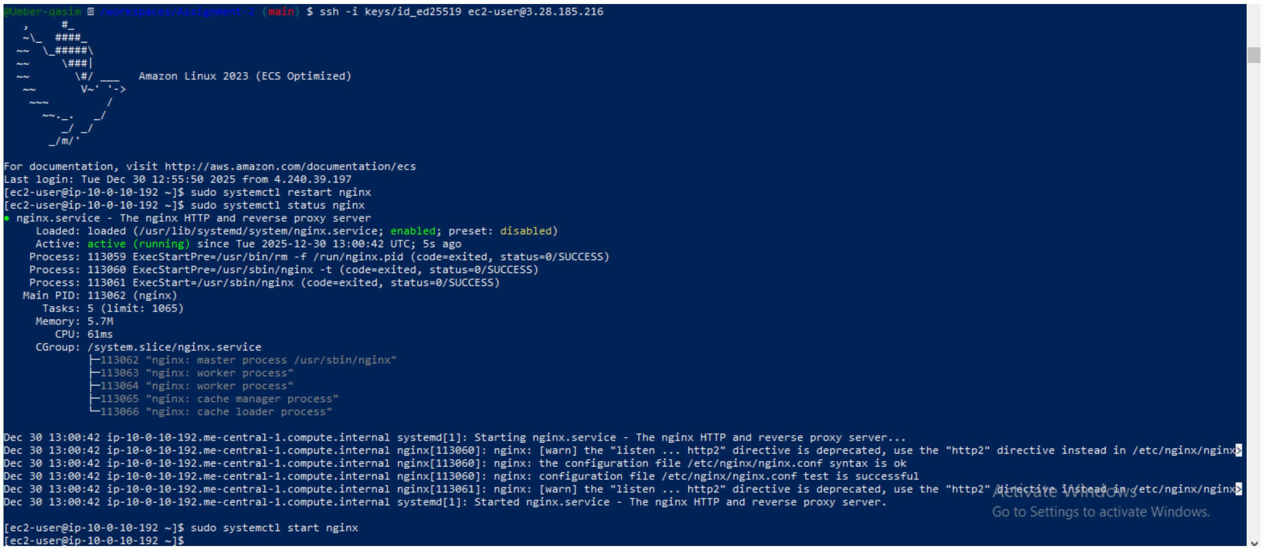
Managed By: Terraform



```
[ec2-user@ip-10-0-10-192 ~]$ sudo tail -f /var/log/nginx/error.log
2025/12/30 12:42:36 [notice] 96652#96652: start cache loader process 96656
2025/12/30 12:42:40 [error] 96653#96653: *3 connect() failed (111: Connection refused) while connecting to upstream, client: 39.58.139.171, server: _, request: "GET / HTTP/2.0", upstream: "htt
p://10.0.10.108:80/", host: "3.28.185.216"
2025/12/30 12:42:40 [warn] 96653#96653: *3 upstream server temporarily disabled while connecting to upstream, client: 39.58.139.171, server: _, request: "GET / HTTP/2.0", upstream: "http://10.
0.10.108:80/", host: "3.28.185.216"
2025/12/30 12:42:40 [error] 96653#96653: *3 connect() failed (111: Connection refused) while connecting to upstream, client: 39.58.139.171, server: _, request: "GET / HTTP/2.0", upstream: "htt
p://10.0.10.57:80/", host: "3.28.185.216"
2025/12/30 12:42:40 [warn] 96653#96653: *3 upstream server temporarily disabled while connecting to upstream, client: 39.58.139.171, server: _, request: "GET / HTTP/2.0", upstream: "http://10.
0.10.57:80/", host: "3.28.185.216"
2025/12/30 12:42:59 [crit] 96653#96653: *7 SSL_do_handshake() failed (SSL: error:0A000172:SSL routines::wrong signature type) while SSL handshaking, client: 66.175.208.116, server: 0.0.0.0:443
2025/12/30 12:43:36 [notice] 96656#96656: http file cache: /var/cache/nginx 0.004M, bsize: 4096
2025/12/30 12:43:36 [notice] 96652#96652: signal 17 (SIGCHLD) received from 96656
2025/12/30 12:43:36 [notice] 96652#96652: cache loader process 96656 exited with code 0
2025/12/30 12:43:36 [notice] 96652#96652: signal 29 (SIGIO) received
```

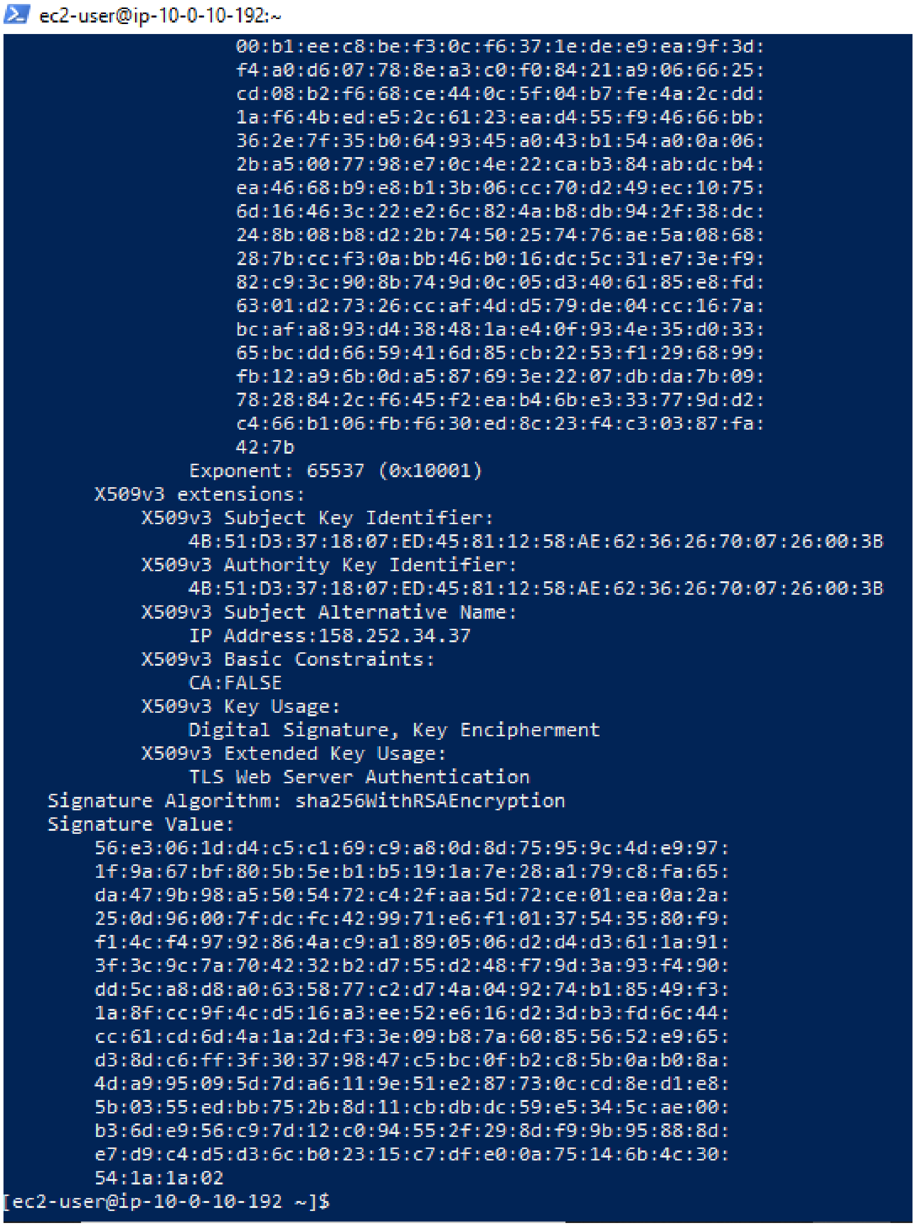


```
[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-30 12:42:36 UTC; 8min ago
    Process: 96648 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 96650 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 96651 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 96652 (nginx)
      Tasks: 4 (limit: 1065)
     Memory: 5.5M
        CPU: 81ms
     CGroup: /system.slice/nginx.service
             ├─96652 "nginx: master process /usr/sbin/nginx"
             ├─96653 "nginx: worker process"
             ├─96654 "nginx: worker process"
             └─96655 "nginx: cache manager process"

Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal nginx[96650]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.>
Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal nginx[96650]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal nginx[96650]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal nginx[96651]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.>
Dec 30 12:42:36 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.
```



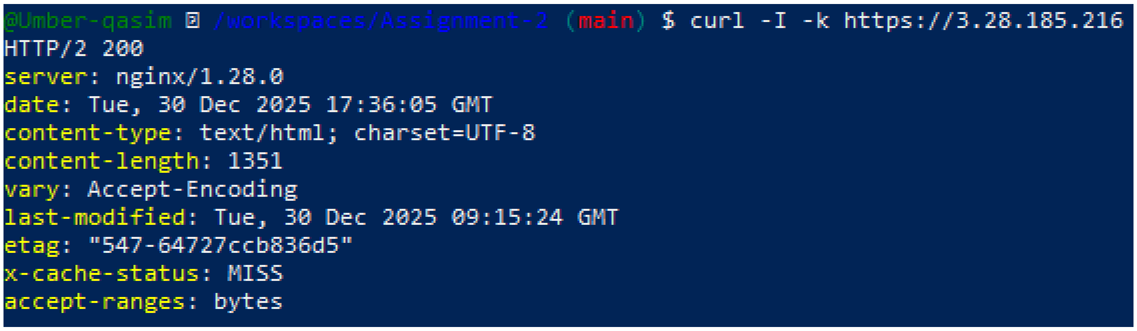
```
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $ ssh -i keys/id_ed25519 ec2-user@3.28.185.216
   ,     #_
   ~\_  ####_
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   Amazon Linux 2023 (ECS Optimized)
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

For documentation, visit http://aws.amazon.com/documentation/ecs
Last login: Tue Dec 30 12:55:50 2025 from 4.240.39.197
[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl status nginx
● nginx.service - The nginx HTTP and reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-30 13:00:42 UTC; 5s ago
    Process: 113059 ExecStartPre=/usr/bin/rm -f /run/nginx.pid (code=exited, status=0/SUCCESS)
    Process: 113060 ExecStartPre=/usr/sbin/nginx -t (code=exited, status=0/SUCCESS)
    Process: 113061 ExecStart=/usr/sbin/nginx (code=exited, status=0/SUCCESS)
   Main PID: 113062 (nginx)
      Tasks: 5 (limit: 1065)
     Memory: 5.7M
        CPU: 61ms
     CGroup: /system.slice/nginx.service
             ├─113062 "nginx: master process /usr/sbin/nginx"
             ├─113063 "nginx: worker process"
             ├─113064 "nginx: worker process"
             ├─113065 "nginx: cache manager process"
             └─113066 "nginx: cache loader process"

Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Starting nginx.service - The nginx HTTP and reverse proxy server...
Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal nginx[113060]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx>
Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal nginx[113060]: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal nginx[113060]: nginx: configuration file /etc/nginx/nginx.conf test is successful
Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal nginx[113061]: nginx: [warn] the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx>
Dec 30 13:00:42 ip-10-0-10-192.me-central-1.compute.internal systemd[1]: Started nginx.service - The nginx HTTP and reverse proxy server.

[ec2-user@ip-10-0-10-192 ~]$ sudo systemctl start nginx
[ec2-user@ip-10-0-10-192 ~]$
```

5.5 Security & Performance Analysis

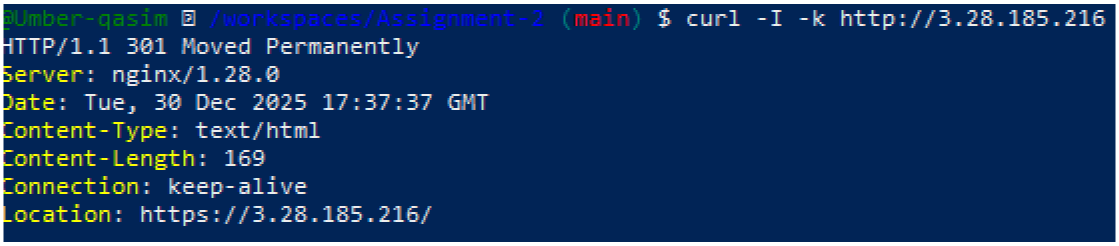


ec2-user@ip-10-0-10-192:~

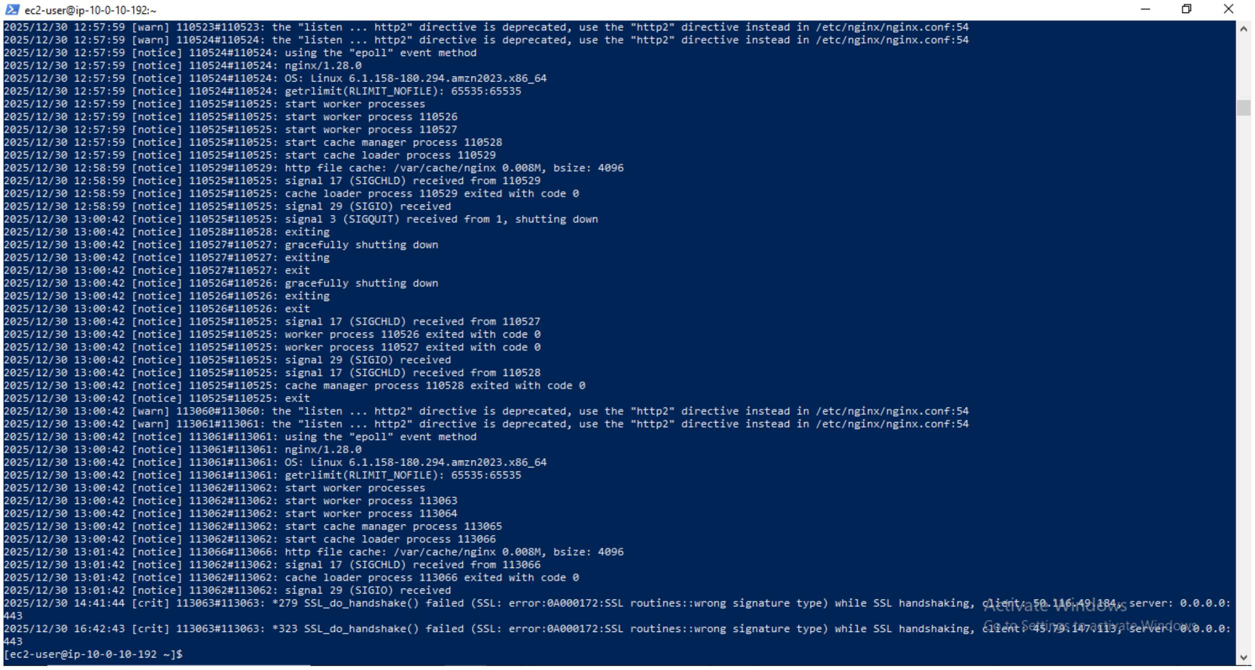
```
                    00:b1:ee:c8:be:f3:0c:f6:37:1e:de:e9:ea:9f:3d:
                    f4:a0:d6:07:78:8e:a3:c0:f0:84:21:a9:06:66:25:
                    cd:08:b2:f6:68:ce:44:0c:5f:04:b7:fe:4a:2c:dd:
                    1a:f6:4b:ed:e5:2c:61:23:ea:d4:55:f9:46:66:bb:
                    36:2e:7f:35:b0:64:93:45:a0:43:b1:54:a0:0a:06:
                    2b:a5:00:77:98:e7:0c:4e:22:ca:b3:84:ab:dc:b4:
                    ea:46:68:b9:e8:b1:3b:06:cc:70:d2:49:ec:10:75:
                    6d:16:46:3c:22:e2:6c:82:4a:b8:db:94:2f:38:dc:
                    24:8b:08:b8:d2:2b:74:50:25:74:76:ae:5a:08:68:
                    28:7b:cc:f3:0a:bb:46:b0:16:dc:5c:31:e7:3e:f9:
                    82:c9:3c:90:8b:74:9d:0c:05:d3:40:61:85:e8:fd:
                    63:01:d2:73:26:cc:af:4d:d5:79:de:04:cc:16:7a:
                    bc:af:a8:93:d4:38:48:1a:e4:0f:93:4e:35:d0:33:
                    65:bc:dd:66:59:41:6d:85:cb:22:53:f1:29:68:99:
                    fb:12:a9:6b:0d:a5:87:69:3e:22:07:db:da:7b:09:
                    78:28:84:2c:f6:45:f2:ea:b4:6b:e3:33:77:9d:d2:
                    c4:66:b1:06:fb:f6:30:ed:8c:23:f4:c3:03:87:fa:
                    42:7b
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                4B:51:D3:37:18:07:ED:45:81:12:58:AE:62:36:26:70:07:26:00:3B
            X509v3 Authority Key Identifier:
                4B:51:D3:37:18:07:ED:45:81:12:58:AE:62:36:26:70:07:26:00:3B
            X509v3 Subject Alternative Name:
                IP Address:158.252.34.37
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Key Usage:
                Digital Signature, Key Encipherment
            X509v3 Extended Key Usage:
                TLS Web Server Authentication
    Signature Algorithm: sha256WithRSAEncryption
    Signature Value:
        56:e3:06:1d:d4:c5:c1:69:c9:a8:0d:8d:75:95:9c:4d:e9:97:
        1f:9a:67:bf:80:5b:5e:b1:b5:19:1a:7e:28:a1:79:c8:fa:65:
        da:47:9b:98:a5:50:54:72:c4:2f:aa:5d:72:ce:01:ea:0a:2a:
        25:0d:96:00:7f:dc:fc:42:99:71:e6:f1:01:37:54:35:80:f9:
        f1:4c:f4:97:92:86:4a:c9:a1:89:05:06:d2:d4:d3:61:1a:91:
        3f:3c:9c:7a:70:42:32:b2:d7:55:d2:48:f7:9d:3a:93:f4:90:
        dd:5c:a8:d8:a0:63:58:77:c2:d7:4a:04:92:74:b1:85:49:f3:
        1a:8f:cc:9f:4c:d5:16:a3:ee:52:e6:16:d2:3d:b3:fd:6c:44:
        cc:61:cd:6d:4a:1a:2d:f3:3e:09:b8:7a:60:85:56:52:e9:65:
        d3:8d:c6:ff:3f:30:37:98:47:c5:bc:0f:b2:c8:5b:0a:b0:8a:
        4d:a9:95:09:5d:7d:a6:11:9e:51:e2:87:73:0c:cd:8e:d1:e8:
        5b:03:55:ed:bb:75:2b:8d:11:cb:db:dc:59:e5:34:5c:ae:00:
        b3:6d:e9:56:c9:7d:12:c0:94:55:2f:29:8d:f9:9b:95:88:8d:
        e7:d9:c4:d5:d3:6c:b0:23:15:c7:df:e0:0a:75:14:6b:4c:30:
        54:1a:1a:02
[ec2-user@ip-10-0-10-192 ~]$
```



```
@Umber-qasim □ /workspaces/Assignment-2 (main) $ curl -I -k https://3.28.185.216
HTTP/2 200
server: nginx/1.28.0
date: Tue, 30 Dec 2025 17:36:05 GMT
content-type: text/html; charset=UTF-8
content-length: 1351
vary: Accept-Encoding
last-modified: Tue, 30 Dec 2025 09:15:24 GMT
etag: "547-64727ccb836d5"
x-cache-status: MISS
accept-ranges: bytes
```

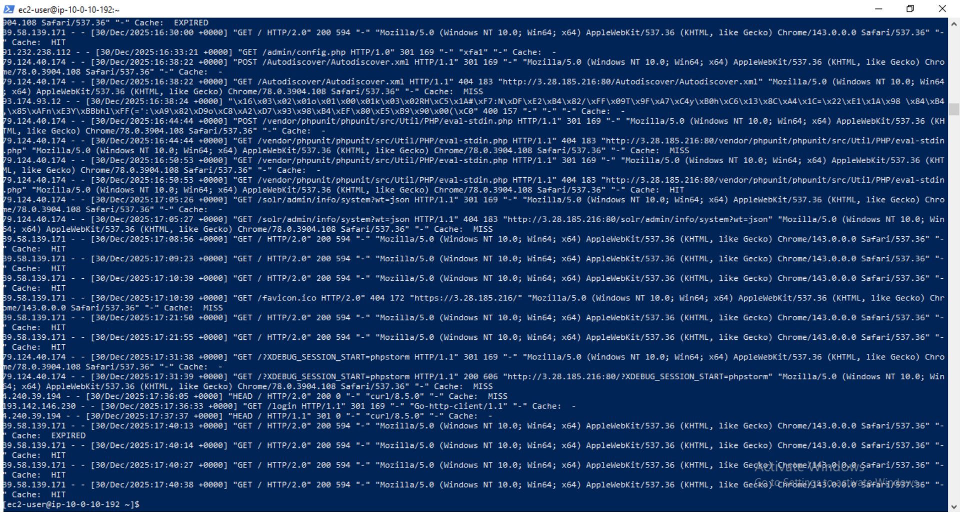


```
@Umber-qasim ▯ /workspaces/Assignment-2 (main) $ curl -I -k http://3.28.185.216
HTTP/1.1 301 Moved Permanently
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 17:37:37 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
Location: https://3.28.185.216/
```



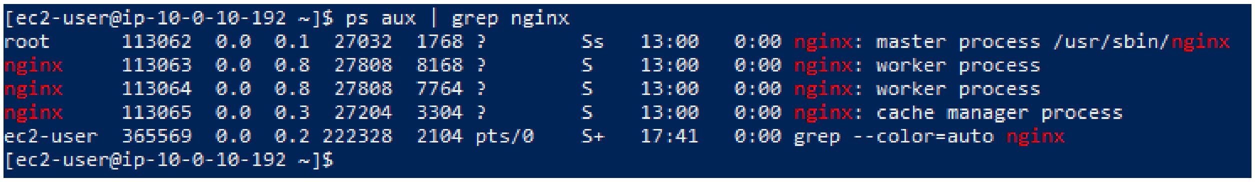
ec2-user@ip-10-0-10-192:~

```
2025/12/30 12:57:59 [warn] 110523#110523: the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
2025/12/30 12:57:59 [warn] 110524#110524: the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
2025/12/30 12:57:59 [notice] 110524#110524: using the "epoll" event method
2025/12/30 12:57:59 [notice] 110524#110524: nginx/1.28.0
2025/12/30 12:57:59 [notice] 110524#110524: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2025/12/30 12:57:59 [notice] 110524#110524: getrlimit(RLIMIT_NOFILE): 65535:65535
2025/12/30 12:57:59 [notice] 110525#110525: start worker processes
2025/12/30 12:57:59 [notice] 110525#110525: start worker process 110526
2025/12/30 12:57:59 [notice] 110525#110525: start worker process 110527
2025/12/30 12:57:59 [notice] 110525#110525: start cache manager process 110528
2025/12/30 12:57:59 [notice] 110525#110525: start cache loader process 110529
2025/12/30 12:58:59 [notice] 110529#110529: http file cache: /var/cache/nginx 0.008M, bsize: 4096
2025/12/30 12:58:59 [notice] 110525#110525: signal 17 (SIGCHLD) received from 110529
2025/12/30 12:58:59 [notice] 110525#110525: cache loader process 110529 exited with code 0
2025/12/30 12:58:59 [notice] 110525#110525: signal 29 (SIGIO) received
2025/12/30 13:00:42 [notice] 110525#110525: signal 3 (SIGQUIT) received from 1, shutting down
2025/12/30 13:00:42 [notice] 110528#110528: exiting
2025/12/30 13:00:42 [notice] 110527#110527: gracefully shutting down
2025/12/30 13:00:42 [notice] 110527#110527: exiting
2025/12/30 13:00:42 [notice] 110527#110527: exit
2025/12/30 13:00:42 [notice] 110526#110526: gracefully shutting down
2025/12/30 13:00:42 [notice] 110526#110526: exiting
2025/12/30 13:00:42 [notice] 110526#110526: exit
2025/12/30 13:00:42 [notice] 110525#110525: signal 17 (SIGCHLD) received from 110527
2025/12/30 13:00:42 [notice] 110525#110525: worker process 110526 exited with code 0
2025/12/30 13:00:42 [notice] 110525#110525: worker process 110527 exited with code 0
2025/12/30 13:00:42 [notice] 110525#110525: signal 29 (SIGIO) received
2025/12/30 13:00:42 [notice] 110525#110525: signal 17 (SIGCHLD) received from 110528
2025/12/30 13:00:42 [notice] 110525#110525: cache manager process 110528 exited with code 0
2025/12/30 13:00:42 [notice] 110525#110525: exit
2025/12/30 13:00:42 [warn] 113060#113060: the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
2025/12/30 13:00:42 [warn] 113061#113061: the "listen ... http2" directive is deprecated, use the "http2" directive instead in /etc/nginx/nginx.conf:54
2025/12/30 13:00:42 [notice] 113061#113061: using the "epoll" event method
2025/12/30 13:00:42 [notice] 113061#113061: nginx/1.28.0
2025/12/30 13:00:42 [notice] 113061#113061: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2025/12/30 13:00:42 [notice] 113061#113061: getrlimit(RLIMIT_NOFILE): 65535:65535
2025/12/30 13:00:42 [notice] 113062#113062: start worker processes
2025/12/30 13:00:42 [notice] 113062#113062: start worker process 113063
2025/12/30 13:00:42 [notice] 113062#113062: start worker process 113064
2025/12/30 13:00:42 [notice] 113062#113062: start cache manager process 113065
2025/12/30 13:00:42 [notice] 113062#113062: start cache loader process 113066
2025/12/30 13:01:42 [notice] 113066#113066: http file cache: /var/cache/nginx 0.008M, bsize: 4096
2025/12/30 13:01:42 [notice] 113062#113062: signal 17 (SIGCHLD) received from 113066
2025/12/30 13:01:42 [notice] 113062#113062: cache loader process 113066 exited with code 0
2025/12/30 13:01:42 [notice] 113062#113062: signal 29 (SIGIO) received
2025/12/30 14:41:44 [crit] 113063#113063: *279 SSL_do_handshake() failed (SSL: error:0A000172:SSL routines::wrong signature type) while SSL handshaking, client: 50.116.49.184, server: 0.0.0.0:
443
2025/12/30 16:42:43 [crit] 113063#113063: *323 SSL_do_handshake() failed (SSL: error:0A000172:SSL routines::wrong signature type) while SSL handshaking, client: 45.79.147.113, server: 0.0.0.0:
443
[ec2-user@ip-10-0-10-192 ~]$
```



ec2-user@ip-10-0-10-192:~

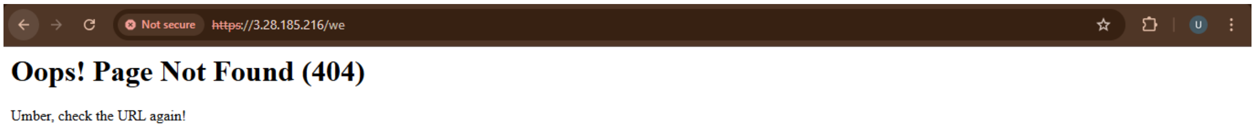
```
904.108 Safari/537.36" "-" Cache:  EXPIRED
39.58.139.171 - - [30/Dec/2025:16:30:00 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
91.232.238.112 - - [30/Dec/2025:16:33:21 +0000] "GET /admin/config.php HTTP/1.0" 301 169 "-" "xfa1" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:16:38:22 +0000] "POST /Autodiscover/Autodiscover.xml HTTP/1.1" 301 169 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chro
me/78.0.3904.108 Safari/537.36" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:16:38:22 +0000] "GET /Autodiscover/Autodiscover.xml HTTP/1.1" 404 183 "http://3.28.185.216:80/Autodiscover/Autodiscover.xml" "Mozilla/5.0 (Windows NT 10.0; Win64
; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  MISS
93.174.93.12 - - [30/Dec/2025:16:38:24 +0000] "\x16\x03\x02\x01o\x01\x00\x01k\x03\x02RH\xC5\x1A#\xF7:N\xDF\xE2\xB4\x82/\xFF\x09T\x9F\xA7\xC4y\xB0h\xC6\x13\x8C\xA4\x1C=\x22\xE1\x1A\x98 \x84\xB4
,\x85\xAFn\xE3Y\xBBbhl\xFF(=':\xA9\x82\xD9o\xC8\xA2\xD7\x93\x98\xB4\xEF\x80\xE5\xB9\x90\x00(\xC0" 400 157 "-" "-" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:16:44:44 +0000] "POST /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 301 169 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KH
TML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:16:44:44 +0000] "GET /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 404 183 "http://3.28.185.216:80/vendor/phpunit/phpunit/src/Util/PHP/eval-stdin
.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  MISS
79.124.40.174 - - [30/Dec/2025:16:50:53 +0000] "GET /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 301 169 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHT
ML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:16:50:53 +0000] "GET /vendor/phpunit/phpunit/src/Util/PHP/eval-stdin.php HTTP/1.1" 404 183 "http://3.28.185.216:80/vendor/phpunit/phpunit/src/Util/PHP/eval-stdin
.php" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  HIT
79.124.40.174 - - [30/Dec/2025:17:05:26 +0000] "GET /solr/admin/info/system?wt=json HTTP/1.1" 301 169 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chro
me/78.0.3904.108 Safari/537.36" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:17:05:27 +0000] "GET /solr/admin/info/system?wt=json HTTP/1.1" 404 183 "http://3.28.185.216:80/solr/admin/info/system?wt=json" "Mozilla/5.0 (Windows NT 10.0; Win
64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  MISS
39.58.139.171 - - [30/Dec/2025:17:08:56 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:09:23 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:10:39 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:10:39 +0000] "GET /favicon.ico HTTP/2.0" 404 172 "https://3.28.185.216/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chr
ome/143.0.0.0 Safari/537.36" "-" Cache:  MISS
39.58.139.171 - - [30/Dec/2025:17:21:50 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:21:55 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
79.124.40.174 - - [30/Dec/2025:17:31:38 +0000] "GET /?XDEBUG_SESSION_START=phpstorm HTTP/1.1" 301 169 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chro
me/78.0.3904.108 Safari/537.36" "-" Cache:  -
79.124.40.174 - - [30/Dec/2025:17:31:39 +0000] "GET /?XDEBUG_SESSION_START=phpstorm HTTP/1.1" 200 606 "http://3.28.185.216:80/?XDEBUG_SESSION_START=phpstorm" "Mozilla/5.0 (Windows NT 10.0; Win
64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.108 Safari/537.36" "-" Cache:  MISS
4.240.39.194 - - [30/Dec/2025:17:36:05 +0000] "HEAD / HTTP/2.0" 200 0 "-" "curl/8.5.0" "-" Cache:  MISS
193.142.146.230 - - [30/Dec/2025:17:36:33 +0000] "GET /login HTTP/1.1" 301 169 "-" "Go-http-client/1.1" "-" Cache:  -
4.240.39.194 - - [30/Dec/2025:17:37:37 +0000] "HEAD / HTTP/1.1" 301 0 "-" "curl/8.5.0" "-" Cache:  -
39.58.139.171 - - [30/Dec/2025:17:40:13 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  EXPIRED
39.58.139.171 - - [30/Dec/2025:17:40:14 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:40:27 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
39.58.139.171 - - [30/Dec/2025:17:40:38 +0000] "GET / HTTP/2.0" 200 594 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36" "-
" Cache:  HIT
[ec2-user@ip-10-0-10-192 ~]$
```



```
[ec2-user@ip-10-0-10-192 ~]$ ps aux | grep nginx
root        113062  0.0  0.1  27032  1768 ?        Ss   13:00   0:00 nginx: master process /usr/sbin/nginx
nginx       113063  0.0  0.8  27808  8168 ?        S    13:00   0:00 nginx: worker process
nginx       113064  0.0  0.8  27808  7764 ?        S    13:00   0:00 nginx: worker process
nginx       113065  0.0  0.3  27204  3304 ?        S    13:00   0:00 nginx: cache manager process
ec2-user    365569  0.0  0.2 222328  2104 pts/0    S+   17:41   0:00 grep --color=auto nginx
[ec2-user@ip-10-0-10-192 ~]$
```

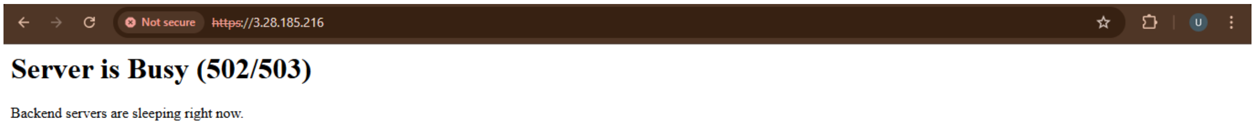
Bonus Tasks

Bonus 1: Custom Error Pages



Oops! Page Not Found (404)

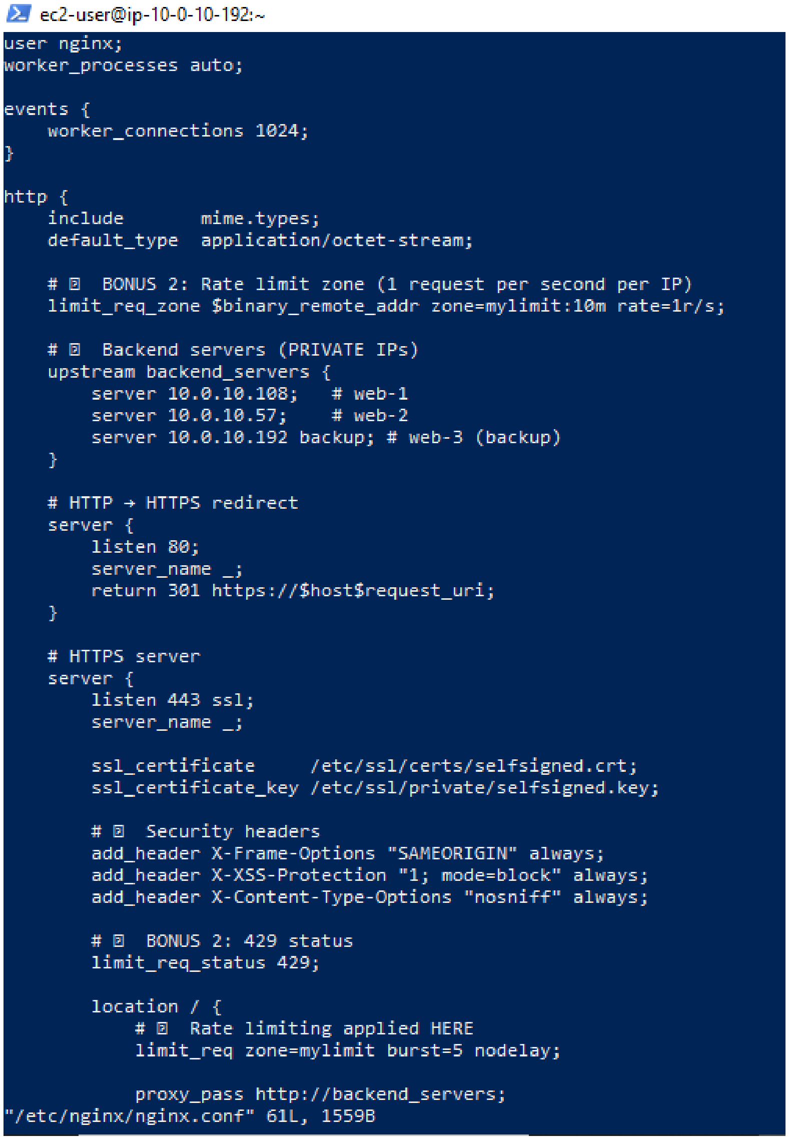
Umber, check the URL again!



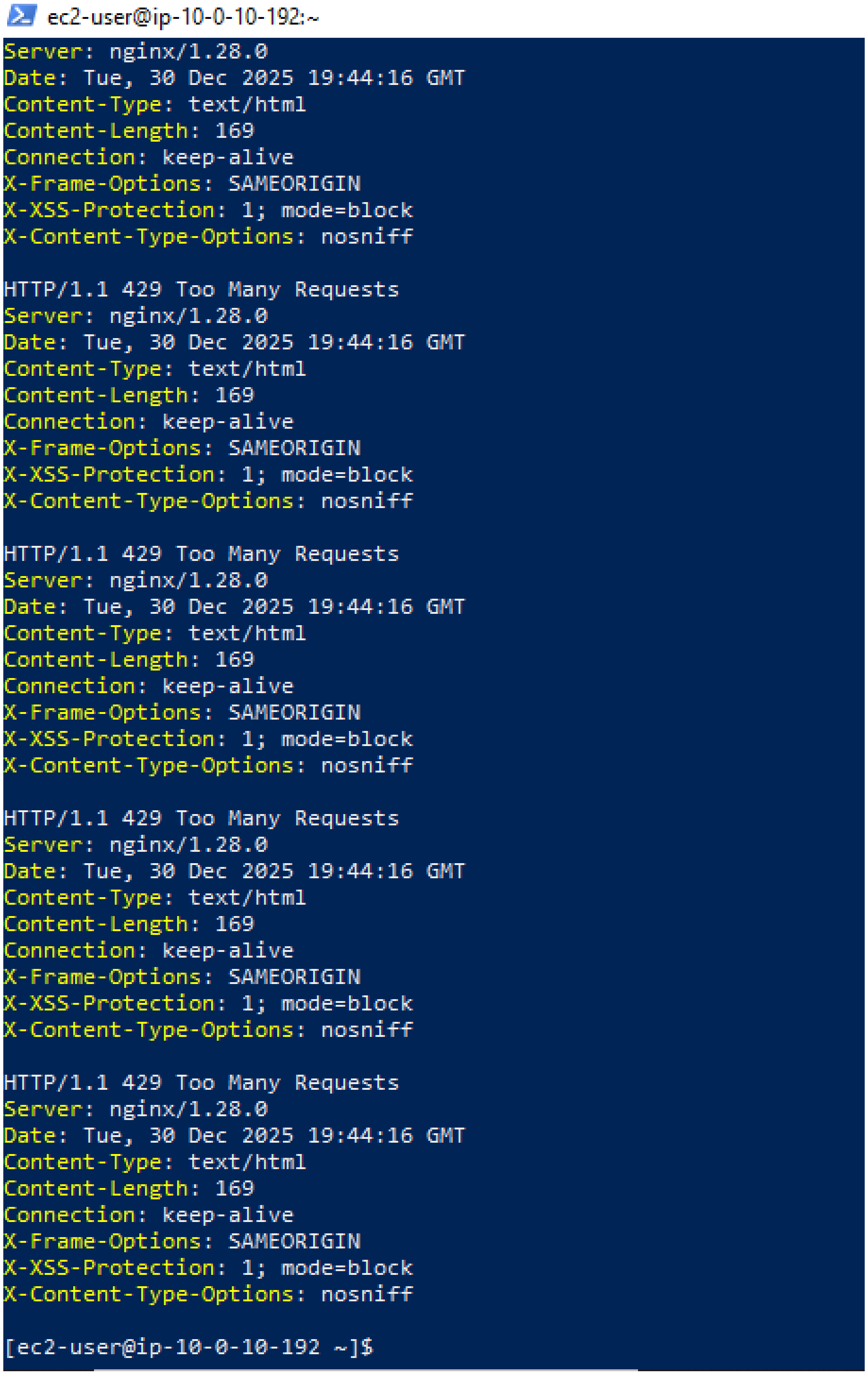
Server is Busy (502/503)

Backend servers are sleeping right now.

Bonus 2: Implement Rate Limiting

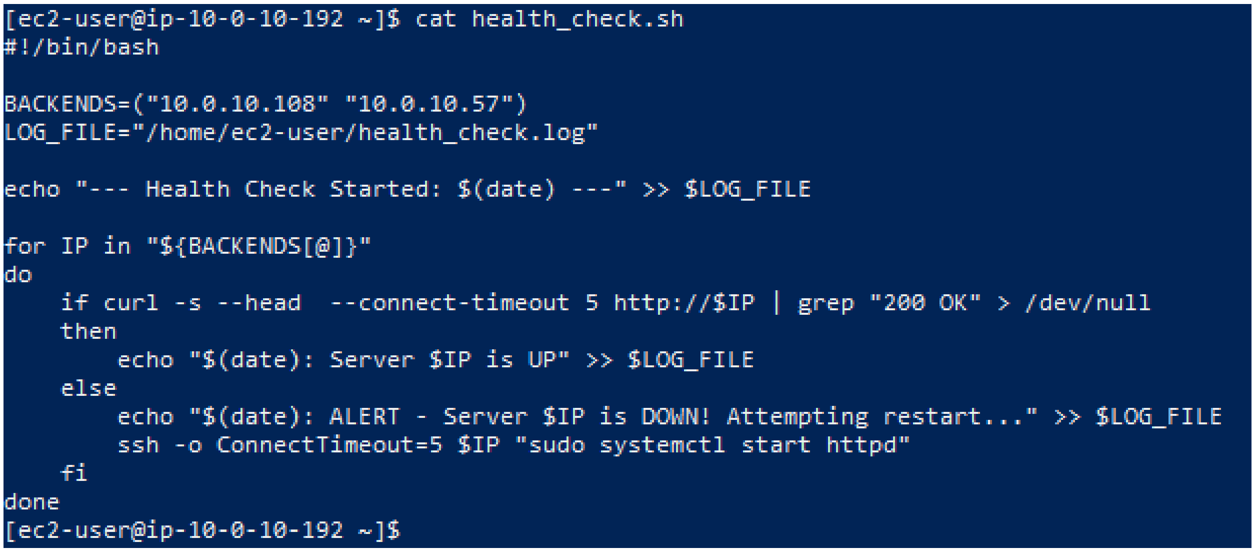


```
user nginx;
worker_processes auto;

events {
    worker_connections 1024;
}

http {
    include       mime.types;
    default_type  application/octet-stream;

    # □  BONUS 2: Rate limit zone (1 request per second per IP)
    limit_req_zone $binary_remote_addr zone=mylimit:10m rate=1r/s;

    # □  Backend servers (PRIVATE IPs)
    upstream backend_servers {
        server 10.0.10.108;   # web-1
        server 10.0.10.57;    # web-2
        server 10.0.10.192 backup; # web-3 (backup)
    }

    # HTTP → HTTPS redirect
    server {
        listen 80;
        server_name _;
        return 301 https://$host$request_uri;
    }

    # HTTPS server
    server {
        listen 443 ssl;
        server_name _;

        ssl_certificate     /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

        # □  Security headers
        add_header X-Frame-Options "SAMEORIGIN" always;
        add_header X-XSS-Protection "1; mode=block" always;
        add_header X-Content-Type-Options "nosniff" always;

        # □  BONUS 2: 429 status
        limit_req_status 429;

        location / {
            # □  Rate limiting applied HERE
            limit_req zone=mylimit burst=5 nodelay;

            proxy_pass http://backend_servers;
"/etc/nginx/nginx.conf" 61L, 1559B
```

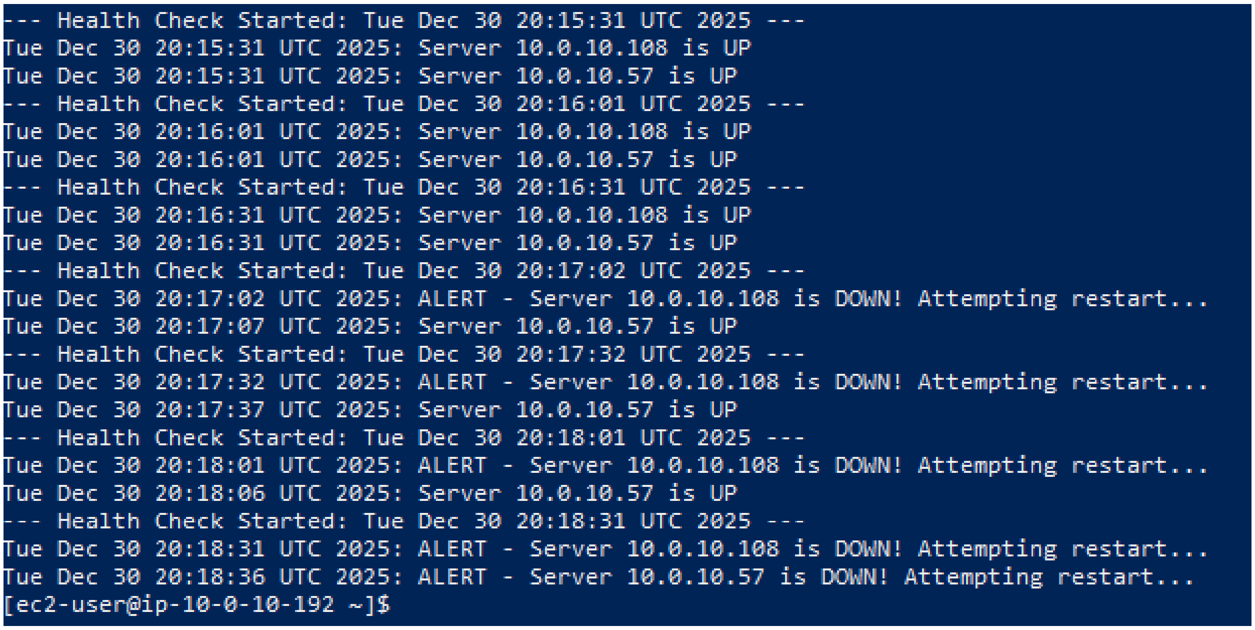


```
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 19:44:16 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

HTTP/1.1 429 Too Many Requests
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 19:44:16 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

HTTP/1.1 429 Too Many Requests
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 19:44:16 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

HTTP/1.1 429 Too Many Requests
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 19:44:16 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

HTTP/1.1 429 Too Many Requests
Server: nginx/1.28.0
Date: Tue, 30 Dec 2025 19:44:16 GMT
Content-Type: text/html
Content-Length: 169
Connection: keep-alive
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff

[ec2-user@ip-10-0-10-192 ~]$
```

Bonus 3: Health Check Automation



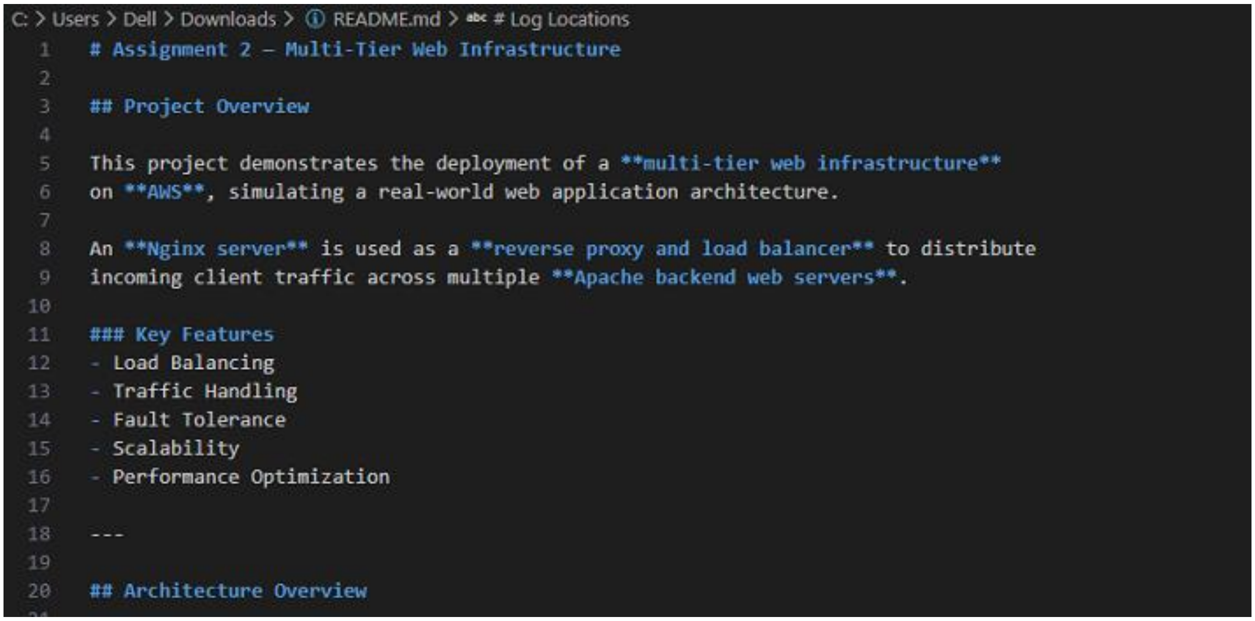
```
[ec2-user@ip-10-0-10-192 ~]$ cat health_check.sh
#!/bin/bash

BACKENDS=("10.0.10.108" "10.0.10.57")
LOG_FILE="/home/ec2-user/health_check.log"

echo "--- Health Check Started: $(date) ---" >> $LOG_FILE

for IP in "${BACKENDS[@]}"
do
    if curl -s --head  --connect-timeout 5 http://$IP | grep "200 OK" > /dev/null
    then
        echo "$(date): Server $IP is UP" >> $LOG_FILE
    else
        echo "$(date): ALERT - Server $IP is DOWN! Attempting restart..." >> $LOG_FILE
        ssh -o ConnectTimeout=5 $IP "sudo systemctl start httpd"
    fi
done
[ec2-user@ip-10-0-10-192 ~]$
```



```
--- Health Check Started: Tue Dec 30 20:15:31 UTC 2025 ---
Tue Dec 30 20:15:31 UTC 2025: Server 10.0.10.108 is UP
Tue Dec 30 20:15:31 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:16:01 UTC 2025 ---
Tue Dec 30 20:16:01 UTC 2025: Server 10.0.10.108 is UP
Tue Dec 30 20:16:01 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:16:31 UTC 2025 ---
Tue Dec 30 20:16:31 UTC 2025: Server 10.0.10.108 is UP
Tue Dec 30 20:16:31 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:17:02 UTC 2025 ---
Tue Dec 30 20:17:02 UTC 2025: ALERT - Server 10.0.10.108 is DOWN! Attempting restart...
Tue Dec 30 20:17:07 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:17:32 UTC 2025 ---
Tue Dec 30 20:17:32 UTC 2025: ALERT - Server 10.0.10.108 is DOWN! Attempting restart...
Tue Dec 30 20:17:37 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:18:01 UTC 2025 ---
Tue Dec 30 20:18:01 UTC 2025: ALERT - Server 10.0.10.108 is DOWN! Attempting restart...
Tue Dec 30 20:18:06 UTC 2025: Server 10.0.10.57 is UP
--- Health Check Started: Tue Dec 30 20:18:31 UTC 2025 ---
Tue Dec 30 20:18:31 UTC 2025: ALERT - Server 10.0.10.108 is DOWN! Attempting restart...
Tue Dec 30 20:18:36 UTC 2025: ALERT - Server 10.0.10.57 is DOWN! Attempting restart...
[ec2-user@ip-10-0-10-192 ~]$
```

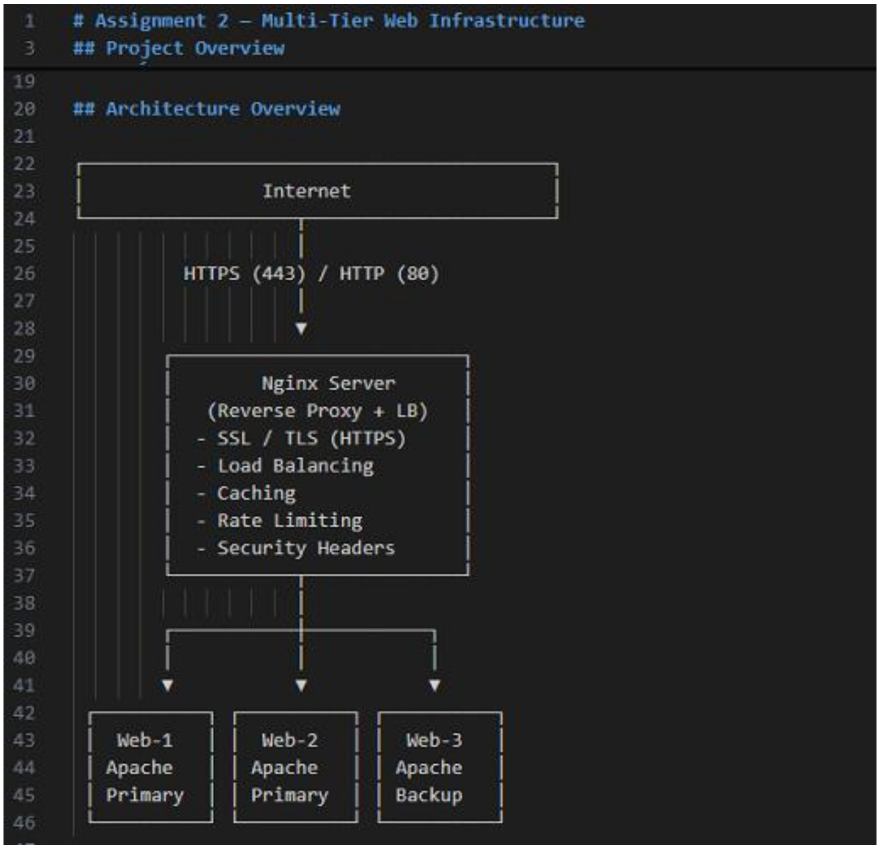
Part 6: Documentation & Cleanup

6.1 README Documentation

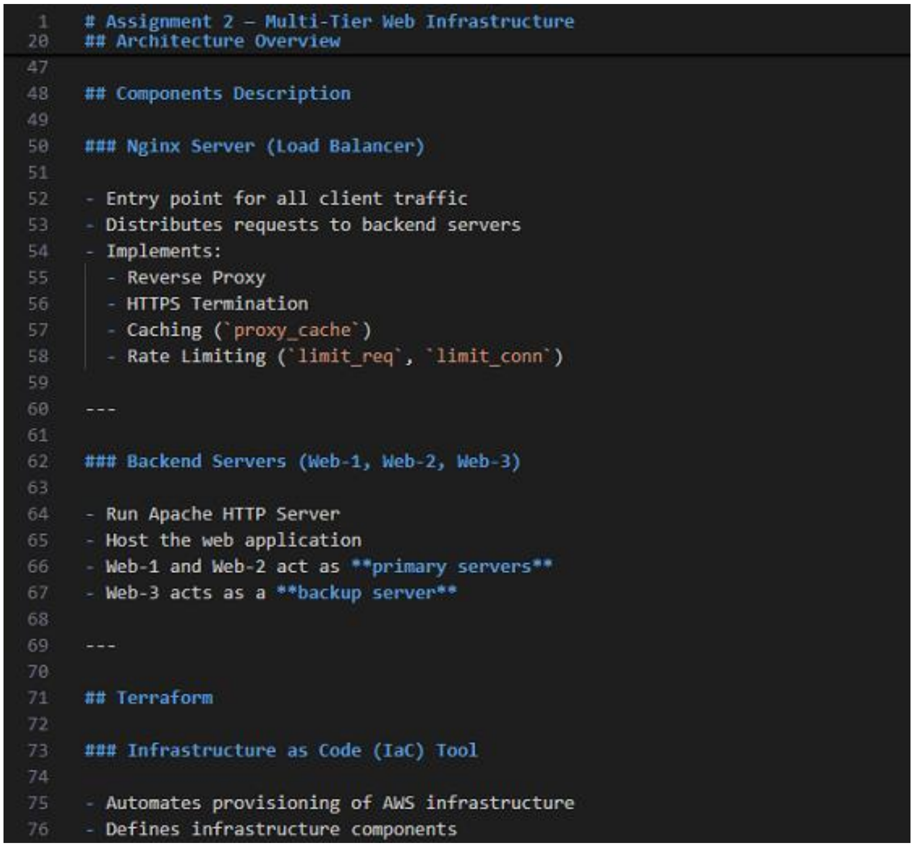


C: > Users > Dell > Downloads > README.md > # Log Locations

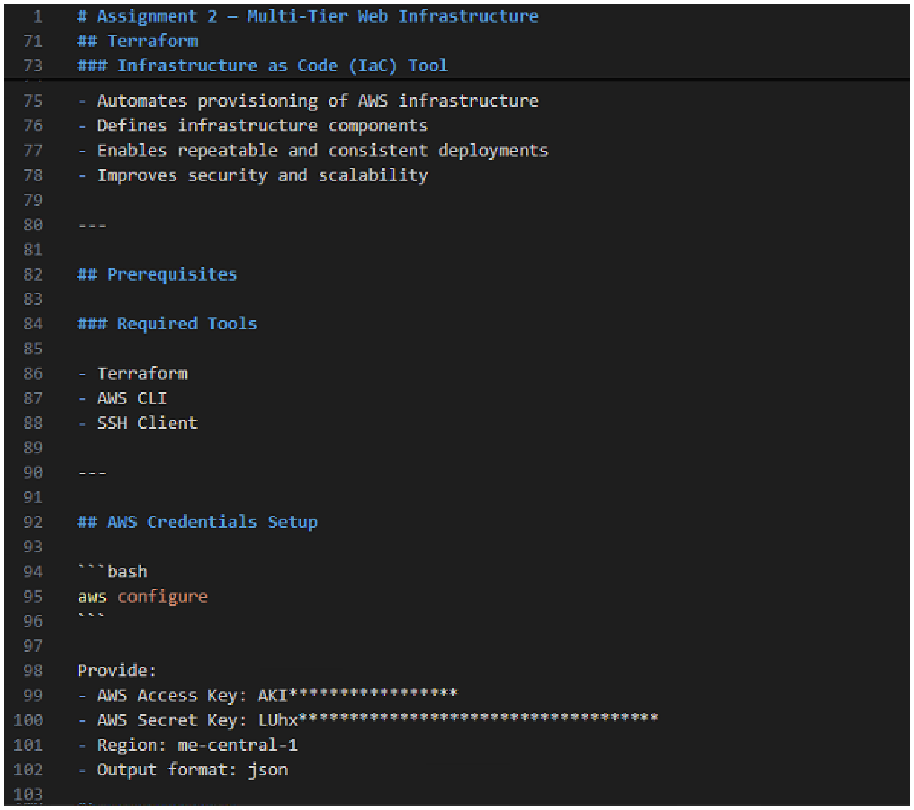
```
# Assignment 2 – Multi-Tier Web Infrastructure

## Project Overview

This project demonstrates the deployment of a **multi-tier web infrastructure**
on **AWS**, simulating a real-world web application architecture.

An **Nginx server** is used as a **reverse proxy and load balancer** to distribute
incoming client traffic across multiple **Apache backend web servers**.

### Key Features
- Load Balancing
- Traffic Handling
- Fault Tolerance
- Scalability
- Performance Optimization

---

## Architecture Overview
```



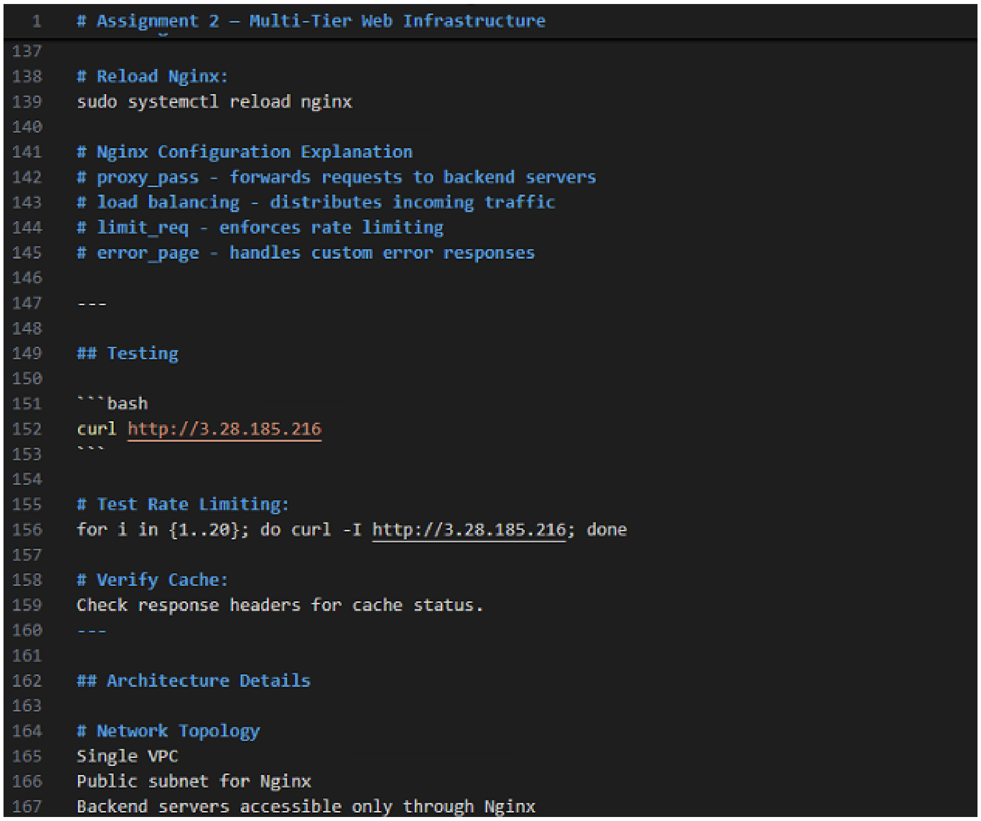
```
# Assignment 2 – Multi-Tier Web Infrastructure
## Project Overview

## Architecture Overview

┌─────────────────────────────────────────┐
│                Internet                 │
└───────────────────┬─────────────────────┘
                    │
          HTTPS (443) / HTTP (80)
                    │
                    ▼
        ┌─────────────────────────┐
        │      Nginx Server       │
        │  (Reverse Proxy + LB)   │
        │  - SSL / TLS (HTTPS)    │
        │  - Load Balancing       │
        │  - Caching              │
        │  - Rate Limiting        │
        │  - Security Headers     │
        └───────────┬─────────────┘
                    │
        ┌───────────┼───────────┐
        │           │           │
        ▼           ▼           ▼
 ┌──────────┐ ┌──────────┐ ┌──────────┐
 │  Web-1   │ │  Web-2   │ │  Web-3   │
 │ Apache   │ │ Apache   │ │ Apache   │
 │ Primary  │ │ Primary  │ │ Backup   │
 └──────────┘ └──────────┘ └──────────┘
```



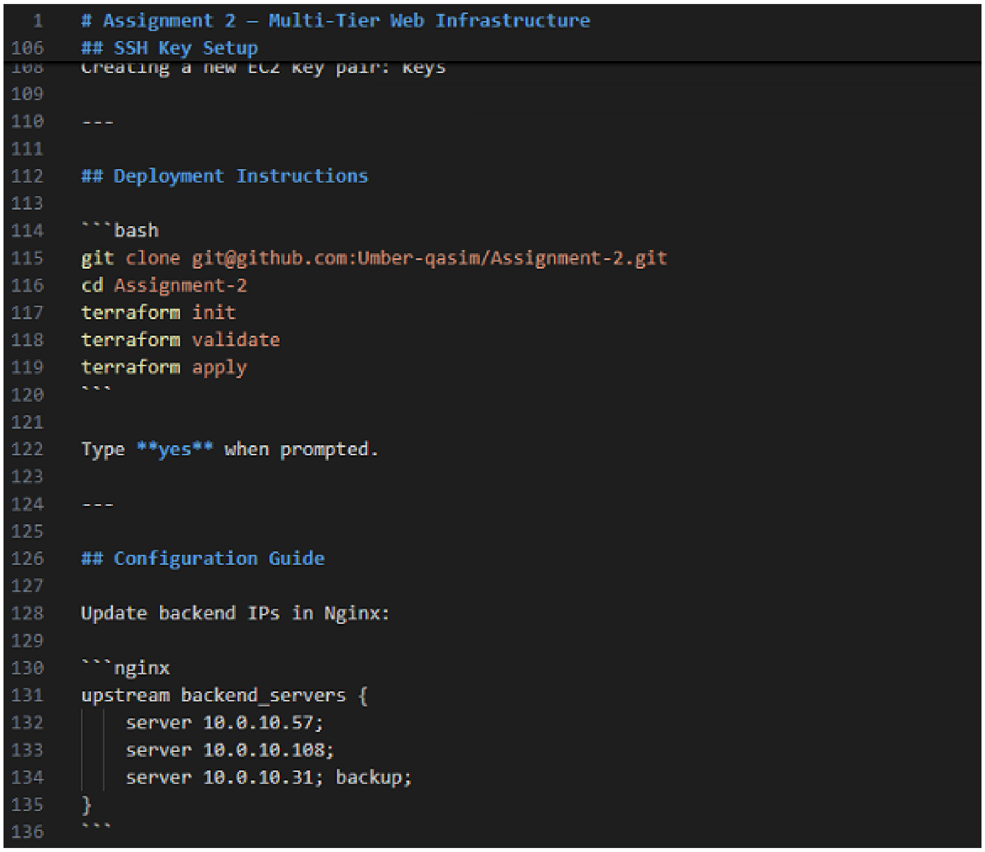
````markdown
# Assignment 2 – Multi-Tier Web Infrastructure
## Architecture Overview

## Components Description

### Nginx Server (Load Balancer)

- Entry point for all client traffic
- Distributes requests to backend servers
- Implements:
  - Reverse Proxy
  - HTTPS Termination
  - Caching (`proxy_cache`)
  - Rate Limiting (`limit_req`, `limit_conn`)

---

### Backend Servers (Web-1, Web-2, Web-3)

- Run Apache HTTP Server
- Host the web application
- Web-1 and Web-2 act as **primary servers**
- Web-3 acts as a **backup server**

---

## Terraform

### Infrastructure as Code (IaC) Tool

- Automates provisioning of AWS infrastructure
- Defines infrastructure components
````



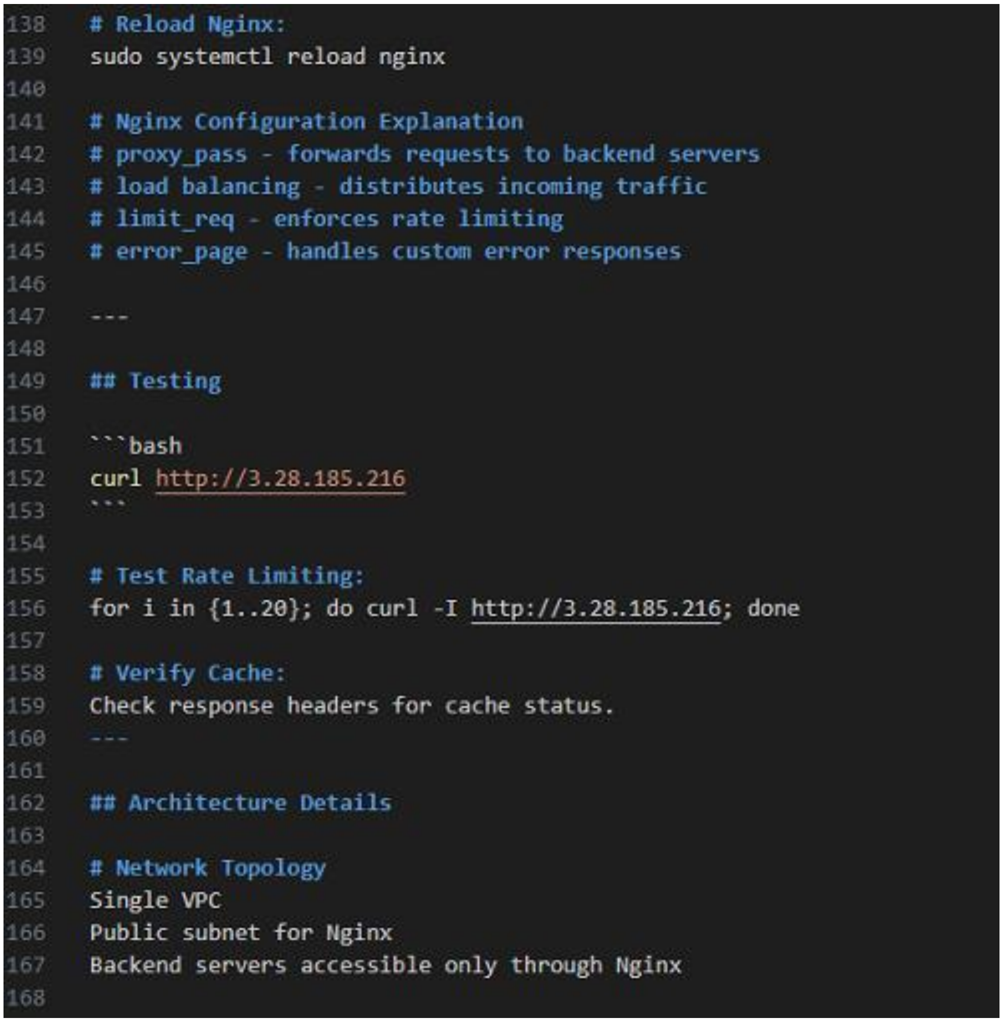
~~~markdown
# Assignment 2 – Multi-Tier Web Infrastructure
## Terraform
### Infrastructure as Code (IaC) Tool
- Automates provisioning of AWS infrastructure
- Defines infrastructure components
- Enables repeatable and consistent deployments
- Improves security and scalability

---

## Prerequisites

### Required Tools

- Terraform
- AWS CLI
- SSH Client

---

## AWS Credentials Setup

```bash
aws configure
```

Provide:
- AWS Access Key: AKI*****************
- AWS Secret Key: LUhx************************************
- Region: me-central-1
- Output format: json

~~~



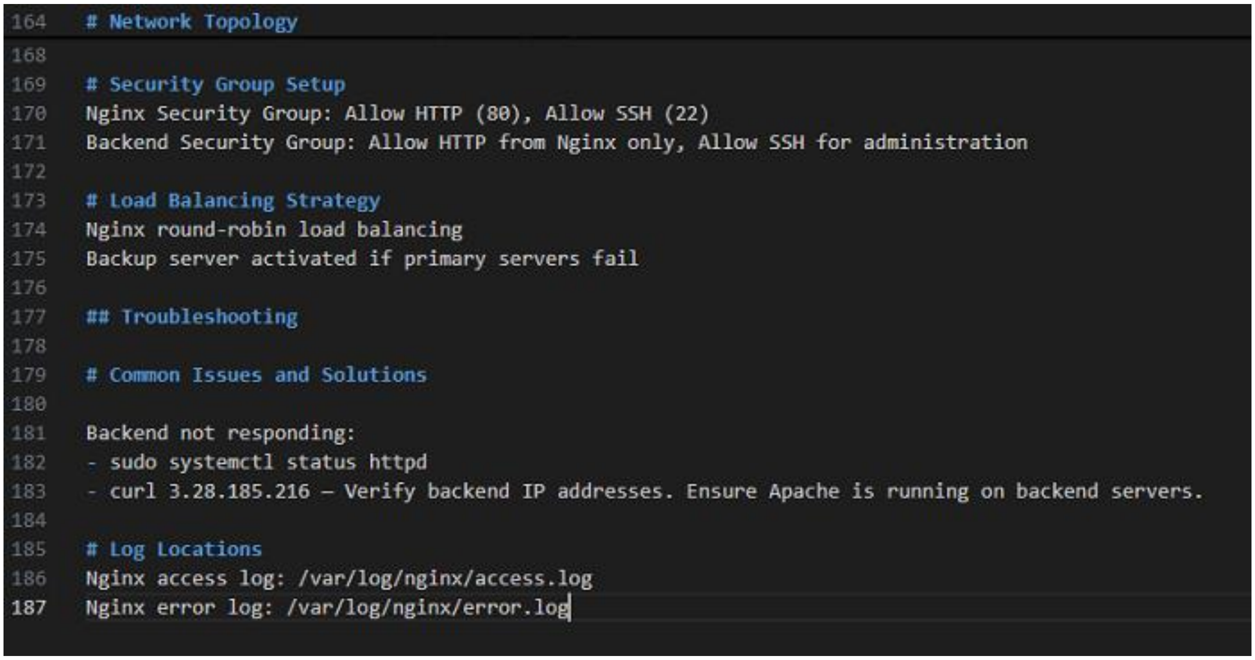
~~~markdown
# Assignment 2 – Multi-Tier Web Infrastructure

# Reload Nginx:
sudo systemctl reload nginx

# Nginx Configuration Explanation
# proxy_pass - forwards requests to backend servers
# load balancing - distributes incoming traffic
# limit_req - enforces rate limiting
# error_page - handles custom error responses

---

## Testing

```bash
curl http://3.28.185.216
```

# Test Rate Limiting:
for i in {1..20}; do curl -I http://3.28.185.216; done

# Verify Cache:
Check response headers for cache status.
---

## Architecture Details

# Network Topology
Single VPC
Public subnet for Nginx
Backend servers accessible only through Nginx
~~~



~~~markdown
# Assignment 2 – Multi-Tier Web Infrastructure
## SSH Key Setup
Creating a new EC2 key pair: keys

---

## Deployment Instructions

```bash
git clone git@github.com:Umber-qasim/Assignment-2.git
cd Assignment-2
terraform init
terraform validate
terraform apply
```

Type **yes** when prompted.

---

## Configuration Guide

Update backend IPs in Nginx:

```nginx
upstream backend_servers {
    server 10.0.10.57;
    server 10.0.10.108;
    server 10.0.10.31; backup;
}
```
~~~

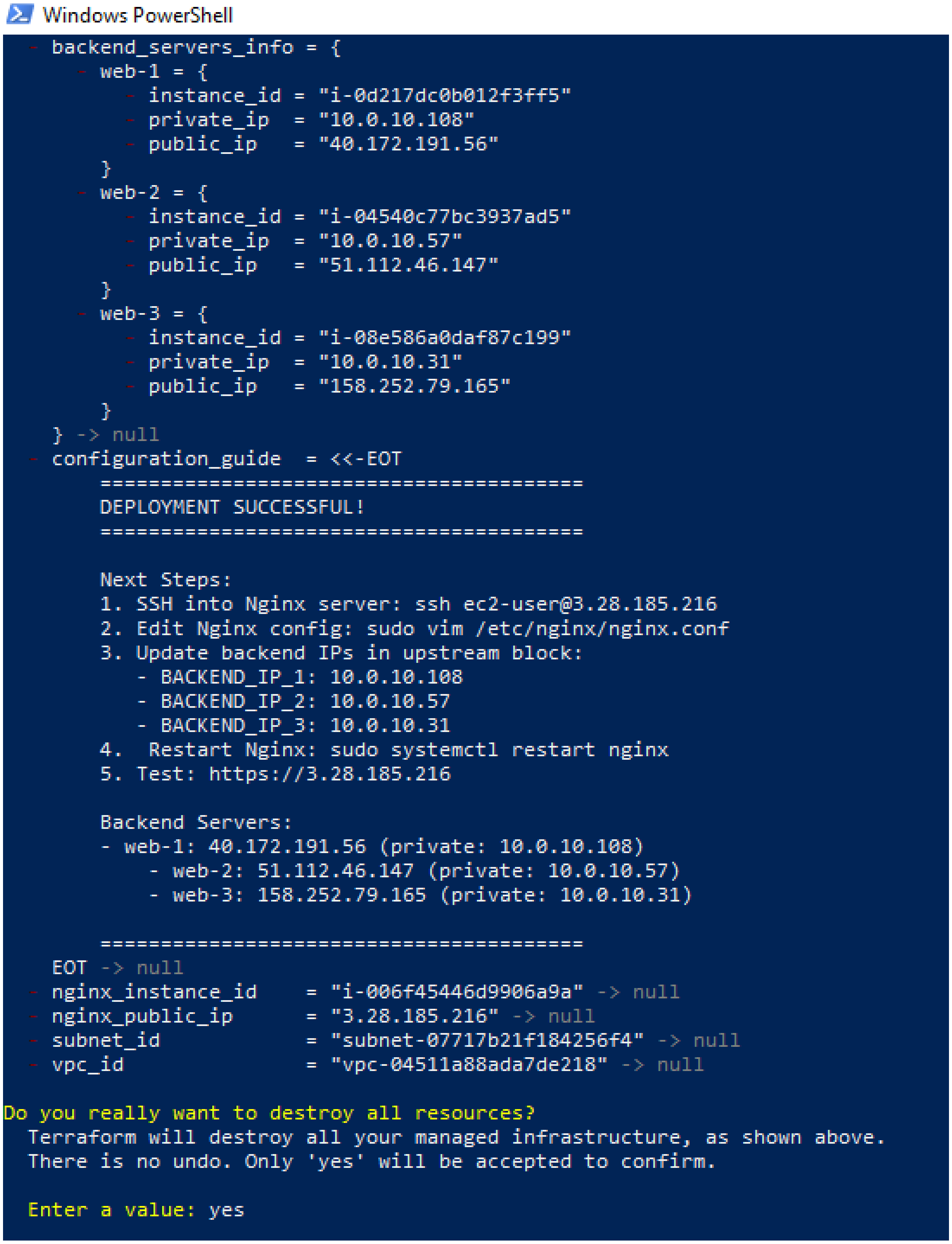


~~~
# Reload Nginx:
sudo systemctl reload nginx

# Nginx Configuration Explanation
# proxy_pass - forwards requests to backend servers
# load balancing - distributes incoming traffic
# limit_req - enforces rate limiting
# error_page - handles custom error responses

---

## Testing

```bash
curl http://3.28.185.216
```

# Test Rate Limiting:
for i in {1..20}; do curl -I http://3.28.185.216; done

# Verify Cache:
Check response headers for cache status.
---

## Architecture Details

# Network Topology
Single VPC
Public subnet for Nginx
Backend servers accessible only through Nginx

~~~

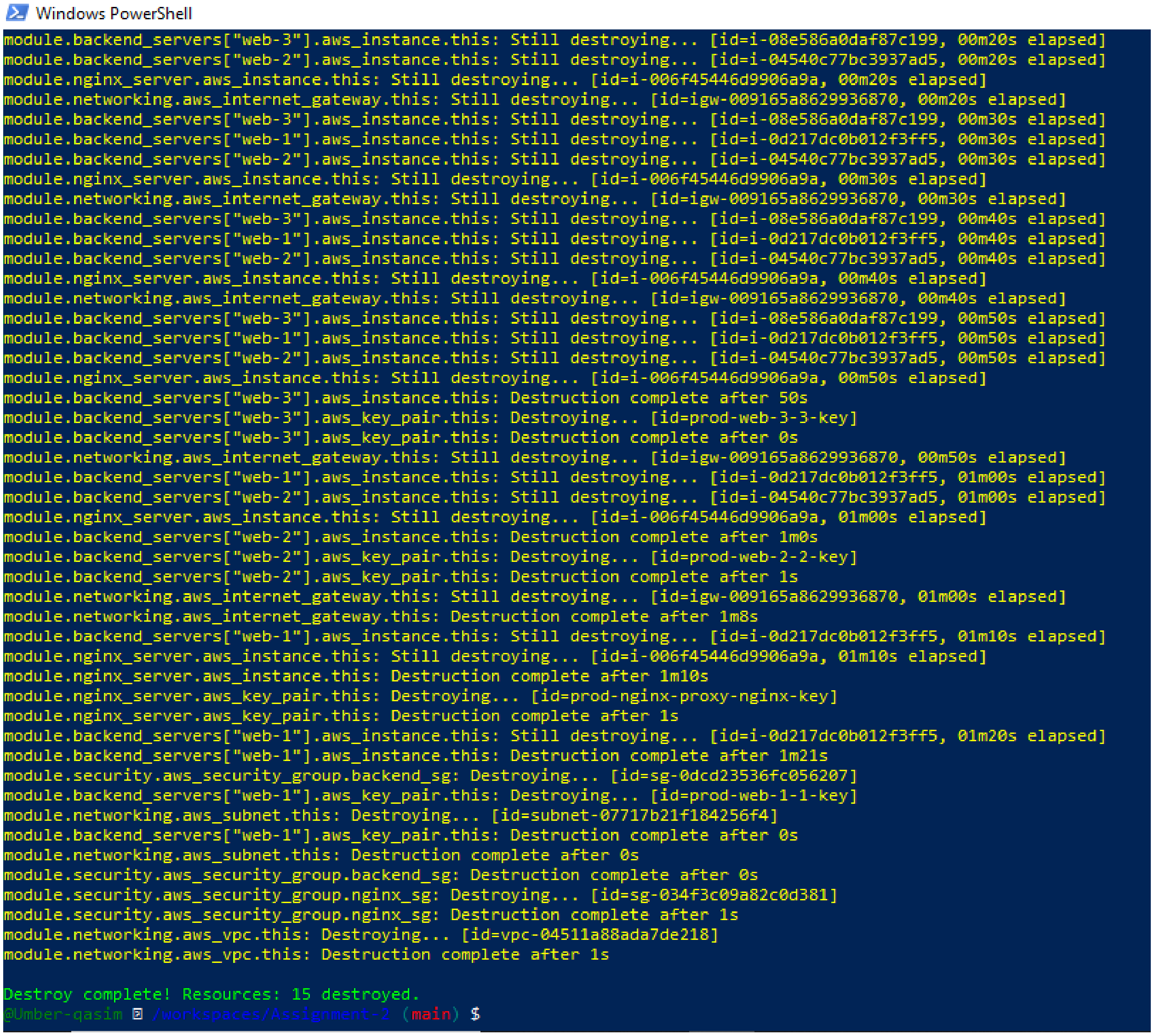


```
# Network Topology

# Security Group Setup
Nginx Security Group: Allow HTTP (80), Allow SSH (22)
Backend Security Group: Allow HTTP from Nginx only, Allow SSH for administration

# Load Balancing Strategy
Nginx round-robin load balancing
Backup server activated if primary servers fail

## Troubleshooting

# Common Issues and Solutions

Backend not responding:
- sudo systemctl status httpd
- curl 3.28.185.216 — Verify backend IP addresses. Ensure Apache is running on backend servers.

# Log Locations
Nginx access log: /var/log/nginx/access.log
Nginx error log: /var/log/nginx/error.log
```

6.2 Infrastructure Cleanup

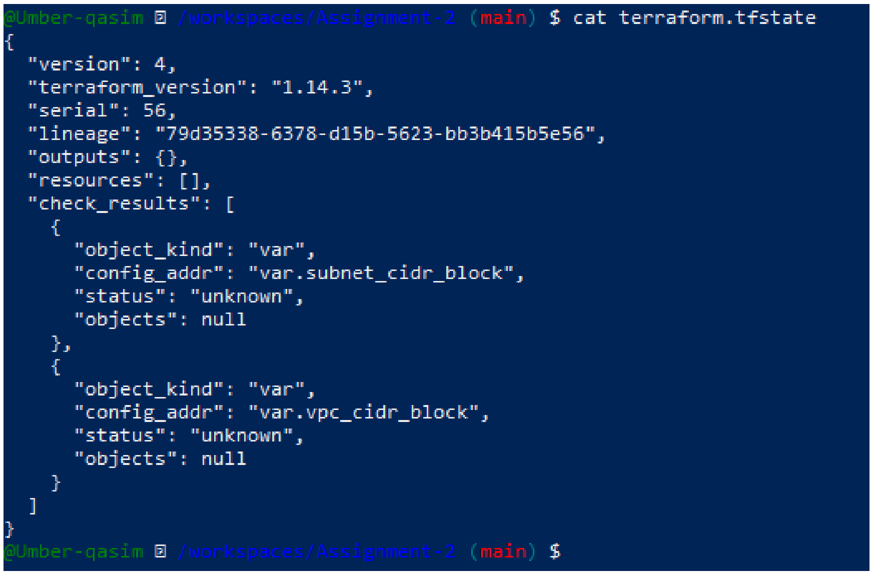


```
  - backend_servers_info = {
      - web-1 = {
          - instance_id = "i-0d217dc0b012f3ff5"
          - private_ip  = "10.0.10.108"
          - public_ip   = "40.172.191.56"
        }
      - web-2 = {
          - instance_id = "i-04540c77bc3937ad5"
          - private_ip  = "10.0.10.57"
          - public_ip   = "51.112.46.147"
        }
      - web-3 = {
          - instance_id = "i-08e586a0daf87c199"
          - private_ip  = "10.0.10.31"
          - public_ip   = "158.252.79.165"
        }
    } -> null
  - configuration_guide  = <<-EOT
        ========================================
        DEPLOYMENT SUCCESSFUL!
        ========================================

        Next Steps:
        1. SSH into Nginx server: ssh ec2-user@3.28.185.216
        2. Edit Nginx config: sudo vim /etc/nginx/nginx.conf
        3. Update backend IPs in upstream block:
           - BACKEND_IP_1: 10.0.10.108
           - BACKEND_IP_2: 10.0.10.57
           - BACKEND_IP_3: 10.0.10.31
        4.  Restart Nginx: sudo systemctl restart nginx
        5. Test: https://3.28.185.216

        Backend Servers:
        - web-1: 40.172.191.56 (private: 10.0.10.108)
            - web-2: 51.112.46.147 (private: 10.0.10.57)
            - web-3: 158.252.79.165 (private: 10.0.10.31)

        ========================================
    EOT -> null
  - nginx_instance_id    = "i-006f45446d9906a9a" -> null
  - nginx_public_ip      = "3.28.185.216" -> null
  - subnet_id            = "subnet-07717b21f184256f4" -> null
  - vpc_id               = "vpc-04511a88ada7de218" -> 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
```

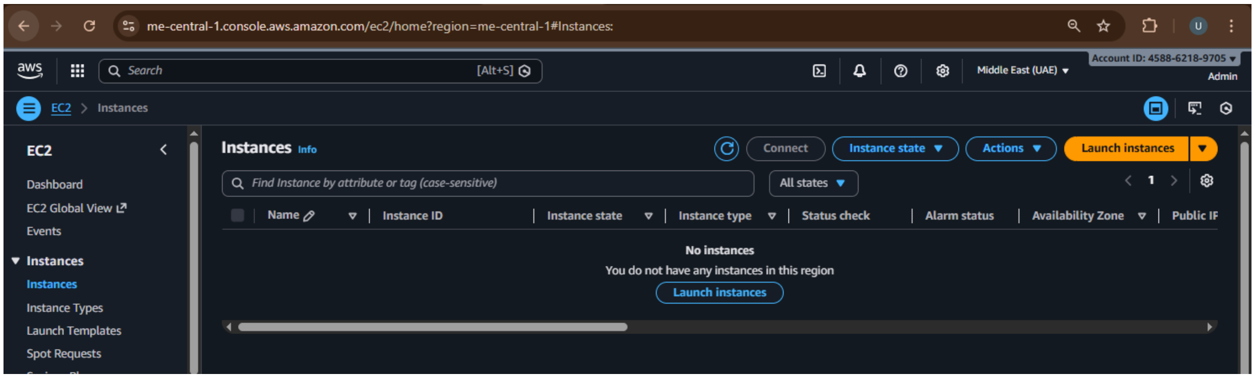


Windows PowerShell

```
module.backend_servers["web-3"].aws_instance.this: Still destroying... [id=i-08e586a0daf87c199, 00m20s elapsed]
module.backend_servers["web-2"].aws_instance.this: Still destroying... [id=i-04540c77bc3937ad5, 00m20s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 00m20s elapsed]
module.networking.aws_internet_gateway.this: Still destroying... [id=igw-009165a8629936870, 00m20s elapsed]
module.backend_servers["web-3"].aws_instance.this: Still destroying... [id=i-08e586a0daf87c199, 00m30s elapsed]
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 00m30s elapsed]
module.backend_servers["web-2"].aws_instance.this: Still destroying... [id=i-04540c77bc3937ad5, 00m30s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 00m30s elapsed]
module.networking.aws_internet_gateway.this: Still destroying... [id=igw-009165a8629936870, 00m30s elapsed]
module.backend_servers["web-3"].aws_instance.this: Still destroying... [id=i-08e586a0daf87c199, 00m40s elapsed]
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 00m40s elapsed]
module.backend_servers["web-2"].aws_instance.this: Still destroying... [id=i-04540c77bc3937ad5, 00m40s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 00m40s elapsed]
module.networking.aws_internet_gateway.this: Still destroying... [id=igw-009165a8629936870, 00m40s elapsed]
module.backend_servers["web-3"].aws_instance.this: Still destroying... [id=i-08e586a0daf87c199, 00m50s elapsed]
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 00m50s elapsed]
module.backend_servers["web-2"].aws_instance.this: Still destroying... [id=i-04540c77bc3937ad5, 00m50s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 00m50s elapsed]
module.backend_servers["web-3"].aws_instance.this: Destruction complete after 50s
module.backend_servers["web-3"].aws_key_pair.this: Destroying... [id=prod-web-3-3-key]
module.backend_servers["web-3"].aws_key_pair.this: Destruction complete after 0s
module.networking.aws_internet_gateway.this: Still destroying... [id=igw-009165a8629936870, 00m50s elapsed]
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 01m00s elapsed]
module.backend_servers["web-2"].aws_instance.this: Still destroying... [id=i-04540c77bc3937ad5, 01m00s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 01m00s elapsed]
module.backend_servers["web-2"].aws_instance.this: Destruction complete after 1m0s
module.backend_servers["web-2"].aws_key_pair.this: Destroying... [id=prod-web-2-2-key]
module.backend_servers["web-2"].aws_key_pair.this: Destruction complete after 1s
module.networking.aws_internet_gateway.this: Still destroying... [id=igw-009165a8629936870, 01m00s elapsed]
module.networking.aws_internet_gateway.this: Destruction complete after 1m8s
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 01m10s elapsed]
module.nginx_server.aws_instance.this: Still destroying... [id=i-006f45446d9906a9a, 01m10s elapsed]
module.nginx_server.aws_instance.this: Destruction complete after 1m10s
module.nginx_server.aws_key_pair.this: Destroying... [id=prod-nginx-proxy-nginx-key]
module.nginx_server.aws_key_pair.this: Destruction complete after 1s
module.backend_servers["web-1"].aws_instance.this: Still destroying... [id=i-0d217dc0b012f3ff5, 01m20s elapsed]
module.backend_servers["web-1"].aws_instance.this: Destruction complete after 1m21s
module.security.aws_security_group.backend_sg: Destroying... [id=sg-0dcd23536fc056207]
module.backend_servers["web-1"].aws_key_pair.this: Destroying... [id=prod-web-1-1-key]
module.networking.aws_subnet.this: Destroying... [id=subnet-07717b21f184256f4]
module.backend_servers["web-1"].aws_key_pair.this: Destruction complete after 0s
module.networking.aws_subnet.this: Destruction complete after 0s
module.security.aws_security_group.backend_sg: Destruction complete after 0s
module.security.aws_security_group.nginx_sg: Destroying... [id=sg-034f3c09a82c0d381]
module.security.aws_security_group.nginx_sg: Destruction complete after 1s
module.networking.aws_vpc.this: Destroying... [id=vpc-04511a88ada7de218]
module.networking.aws_vpc.this: Destruction complete after 1s

Destroy complete! Resources: 15 destroyed.
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $
```



```
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 56,
  "lineage": "79d35338-6378-d15b-5623-bb3b415b5e56",
  "outputs": {},
  "resources": [],
  "check_results": [
    {
      "object_kind": "var",
      "config_addr": "var.subnet_cidr_block",
      "status": "unknown",
      "objects": null
    },
    {
      "object_kind": "var",
      "config_addr": "var.vpc_cidr_block",
      "status": "unknown",
      "objects": null
    }
  ]
}
@Umber-qasim ➜ /workspaces/Assignment-2 (main) $
```



me-central-1.console.aws.amazon.com/ec2/home?region=me-central-1#Instances:

Account ID: 4588-6218-9705

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IF

No instances

You do not have any instances in this region

Challenges & Solutions

Challenge 1: SSH Key Errors

Solution: Regenerated key pairs and updated Terraform variables

Challenge 2: Backend Not Reachable

Solution: Corrected security group rules

Challenge 3: Rate Limiting Not Triggering

Solution: Reduced request rate threshold and tested using loops

Lessons Learned

- Importance of IaC
- Security-first cloud design
- Debugging cloud networking issues

Conclusion

This assignment successfully demonstrated the deployment of a production-like multi-tier architecture using AWS and Terraform.

It enhanced practical skills in:

- Cloud infrastructure
- Linux server automation
- Security and performance optimization
- Troubleshooting real-world issues

Future Improvements

- Use ACM certificates instead of self-signed
- Auto Scaling Groups
- Monitoring with CloudWatch
- CI/CD integration

Appendices

Included

- Complete Terraform code
- Apache and Nginx scripts
- Configuration files
- Screenshots

References:

- AWS Documentation
- Terraform Documentation
- Nginx Official Docs