
LAB 12

NAME: SHUMAIL ZAHRA
REGISTRATION #: 2023-BSE-061
DEPARTMENT: BSE(5B)

LAB TASK

Lab 12 – Terraform Provisioners, Modules & Nginx Reverse Proxy/Load Balancer

Task 0 Lab Setup (Codespace & GH CLI)

task0_codespace_create_and_list

```
Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
host? HTTPste Git with your GitHub credentials? Yes

? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this
host
? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this
host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with
a web browser

! First copy your one-time code: 0E23-B744
Press Enter to open https://github.com/login/device in your
browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as 23-22411-061-rgb
! You were already logged in to this account

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh auth status
github.com
  ✓ Logged in to github.com account 23-22411-061-rgb (keyrin
g)
  - Active account: true
  - Git operations protocol: https
  - Token: gho_*****
  - Token scopes: 'gist', 'read:org', 'repo', 'workflow'

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh repo create CC_ShumailZahra_2023-BSE-061/Lab12 --public
HTTP 404: Not Found (https://api.github.com/users/CC_Shumail
Zahra_2023-BSE-061)

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh repo create CC_ShumailZahra_2023-BSE-061_Lab12 --public
✓ Created repository 23-22411-061-rgb/CC_ShumailZahra_2023-BSE-061_Lab12 on github.com
```

```
✓ Created repository 23-22411-061-rgb/cc_Shumailzahra_2023-BSE-061_Lab12 on github.com
https://github.com/23-22411-061-rgb/cc_Shumailzahra_2023-BSE-061_Lab12

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace create --repo 23-22411-061-rgb/cc_ShumailZahra_2023-BSE-061_Lab12
error getting devcontainer.json paths: HTTP 400: The 'ref' provided was invalid. Please specify a valid branch name or commit SHA (https://api.github.com/repositories/1128160895/codespaces/devcontainers?per_page=100&ref=main)
This API operation needs the "codespace" scope. To request it, run: gh auth refresh -h github.com -s codespace

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ GH CODESPACE LIST
unknown command "CODESPACE" for "gh"

Did you mean this?
  codespace

Usage: gh <command> <subcommand> [flags]

Available commands:
  agent-task
  alias
  api
  attestation
  auth
  browse
  cache
  co
  codespace
  completion
  config
  extension
  gist
? Authenticate Git with your GitHub credentials? Yes
? Authenticate Git with your GitHub credentials? (Y/n) Y
! First copy your one-time code: 947B-128B
Press Enter to open https://github.com/login/device in your browser...
✓ Authentication complete.

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh auth status
github.com
  ✓ Logged in to github.com account 23-22411-061-rgb (keyring)
  - Active account: true
  - Git operations protocol: https
  - Token: gho_***** (REDACTED)
  - Token scopes: 'codespace', 'gist', 'read:org', 'repo', 'workflow'
```

```
- Git operations protocol: https
- Token: gho_*****
- Token scopes: 'codespace', 'gist', 'read:org', 'repo', 'workflow'

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace list
NAME          DISPLAY NAME      REPOSITORY      BRANCH  STATE    CREATED AT
special-space... special space... 23-22411-061-... main* Shutdown about 1 month...
supreme-barnac... supreme barnacle 23-22411-061-... main* Shutdown about 11 days...
glorious-tribb... glorious tribble 23-22411-061-... main   Shutdown about 11 hour...

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace start -c glorious tribble
unknown shorthand flag: 'c' in -c

Usage: gh codespace [flags]

Available commands:
  code
  cp
  create
  delete
  edit
  jupyter
  list
  logs
  ports
  rebuild
  ssh
  stop
  view

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace list --json name,displayName,repository,state
[{"name": "special-space-funicular", "repository": "23-22411-061-rgb/CC_-Shumail-zahra-_-2023-BSE-061-", "state": "Shutdown", "displayName": "special space funicular"}, {"name": "supreme-barnacle-x5g5x469g76wf6qqq", "repository": "23-22411-061-rgb/CC_-Shumail-zahra-_-2023-BSE-061-", "state": "Shutdown", "displayName": "supreme barnacle"}]
```

```

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace start glorious-tribble-wrqr469p7pq29pj
unknown command "start" for "gh codespace"

Usage: gh codespace [flags]

Available commands:
  code
  cp
  create
  delete
  edit
  jupyter
  list
  logs
  ports
  rebuild
  ssh
  stop
  view

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace ssh -c glorious-tribble-wrqr469p7pq29pj
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

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

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

@23-22411-061-rgb → /workspaces/CC_Shumail-zahra--2023-BSE-061- (main) $ exit
logout
Connection to localhost closed.

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh repo list --limit 20

Showing 11 of 11 repositories in @23-22411-061-rgb


```

NAME	DESCRIPTION	INFO	UPDATED
23-22411-061-rgb/CC_ShumailZahra_2023-BSE-061_Lab12		public	about 18 minutes ago
23-22411-061-rgb/CC_23-22411-061-Assigment-2		public	about 3 days ago
23-22411-061-rgb/cc_shumailzahra_2023-BSE-061_Lab11		public	about 11 days ago
23-22411-061-rgb/UbuntuMachine		public, fork	about 2 months ago
23-22411-061-rgb/CC_Shumail-zahra--2023-BSE-061-		public	about 2 months ago
23-22411-061-rgb/23-22411-061-rgb.github.io	Gitea Tutorial	public	about 2 months ago
23-22411-061-rgb/Gitea		public, fork	about 2 months ago
23-22411-061-rgb/CC_Shumailzahra.github.io		public	about 2 months ago
23-22411-061-rgb/gitea-lab		public	about 2 months ago
23-22411-061-rgb/lab-2		private	about 3 months ago

```

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ gh codespace create --repo 23-22411-061-rgb/cc_ShumailZahra_2023-BSE-061_Lab12
  / Codespace usage for this repository is paid for by 23-22411-061-rgb
error: getting devcontainer.json paths: HTTP 400: The 'ref' provided was invalid. Please specify a valid branch name or commit SHA (https://api.github.com/repos/112816095/codespaces/devcontainers?per_page=100&ref=main)

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ git clone https://github.com/23-22411-061-rgb/CC_ShumailZahra_2023-BSE-061_Lab12.git
Cloning into 'CC_ShumailZahra_2023-BSE-061_Lab12'...
warning: You appear to have cloned an empty repository.

Syed@DESKTOP-S50GK51 MINGW64 ~ (main)
$ cd CC_ShumailZahra_2023-BSE-061_Lab12

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ echo "# Lab 12 - Codespace Setup" > README.md

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ git commit -m "Initial commit for Lab 12"
[main (root-commit) 81e7ef9] Initial commit for Lab 12
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

```

```

warning: in the working copy of 'README.md', EOL will be replaced by CR LF the next time git touches it.

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ git commit -m "Initial commit for Lab 12"
[main (root-commit) 81e7ef9] Initial commit for Lab 12
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), 257 bytes | 51.00 KiB/s, done.
Writing objects: 100% (3/3), 257 bytes | 51.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/23-22411-061-rgb/CC_ShumailZahra_2023-BSE-061_Lab12.git
 * [new branch]      main -> main

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ gh codespace create --repo 23-22411-061-rgb/CC_ShumailZahra_2023-BSE-061_Lab12
  ⚠️ A license for this repository is paid for by 23-22411-061-rgb
? Choose Machine Type: 2 cores, 8 GB RAM, 32 GB storage
fantastic-space-train-r454wx9r5j793pqv7

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ gh codespace list
NAME          DISPLAY NAME           REPOSITORY          BRANCH   STATE    CREATED AT
special-space-funicular-97g7wxvqr97x2pxgr special space funicular 23-22411-061-rgb/CC_Shumail-zahra--_2023-BS... main* Shutdown about 1 month ago
supreme-barnacle-x5g5x469g7wf6qq supreme barnacle 23-22411-061-rgb/CC_Shumail-zahra--_2023-BS... main* Shutdown about 11 days ago
glorious-tribble-wqrqx469p7pq23pjj glorious tribble 23-22411-061-rgb/CC_Shumail-zahra--_2023-BS... main Available about 11 hours ago

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

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $ pwd
/wkspces/CC_ShumailZahra_2023-BSE-061_Lab12
@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $
@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $ Read from remote host localhost: Connection reset by peer
Connection to localhost closed.
client_loop: send disconnect: Connection reset by peer

```

task0_codespace_ssh_connected

```

Syed@DESKTOP-S50GK51 MINGW64 ~/CC_ShumailZahra_2023-BSE-061_Lab12 (main)
$ gh codespace ssh -c fantastic-space-train-r454wx9r5j793pqv7
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro
Last login: Mon Jan  5 08:47:34 2026 from ::1
@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $ |

```

Task 1 — Organize Terraform code into separate files

task1_project_directory

```

Last login: Mon Jan  5 08:47:34 2026 from ::1
@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $ mkdir -p ~/Lab12
@23-22411-061-rgb ~ /workspaces/CC_ShumailZahra_2023-BSE-061_Lab12 (main) $ cd ~/Lab12

```

task1_files_created

```

dash: @23-22411-061-rgb: command not found
@23-22411-061-rgb ~ /Lab12 $ touch main.tf variables.tf outputs.tf locals.tf terraform.tfvars entry-script.sh
@23-22411-061-rgb ~ /Lab12 $ |

```

task1_variables_tf

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

~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~
```

task1_outputs_tf

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

```
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~
```

task1_locals_tf

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

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

```
~  
~  
~  
~  
~  
~  
~  
~  
~  
~
```

task1_terraform_tfvars

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

```
~  
~  
~  
~  
~  
~
```

task1_main_tf

```
ingress {
    from_port   = 80
    to_port     = 80
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}
egress {
    from_port   = 0
    to_port     = 0
    protocol   = "-1"
    cidr_blocks = ["0.0.0.0/0"]
    prefix_list_ids = []
}
tags = {
    Name = "${var.env_prefix}-default-sg"
}
}

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

resource "aws_instance" "myapp-server" {
    ami           = "ami-05524d6658fcf35b6" # Amazon Linux 2023 Kernel 6.1 AMI
    instance_type = var.instance_type
    subnet_id    = aws_subnet.myapp_subnet_1.id
    security_groups = [aws_default_security_group.default_sg. id]
    availability_zone = var.availability_zone
    associate_public_ip_address = true
    key_name = aws_key_pair.ssh-key. key_name

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

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

task1_entry_script

```
codespace@codespaces-6aec8: vim entry-script.sh
```

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

task1_ssh_keygen

```
@23-22411-061-rgb ~/Lab12 $ ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -N ""
Generating public/private ed25519 key pair.
Your identification has been saved in /home/codespace/.ssh/id_ed25519
Your public key has been saved in /home/codespace/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:xUgX+WOKR/YeIWeqOTacSS6js0LVhus88UA6r3b1qPM codespace@codespaces-6aec8
The key's randomart image is:
++-[ED25519 256]--+
|       . oo
|       . +.
|       . o.
|       o   .+ B
| * o   S+ O o
|o B o. o + o
|.+ *+ . B . .
|oo+o...o X   .
|o+*E.. =.o
+---[SHA256]----+
@23-22411-061-rgb ~/Lab12 $ |
```

task1_terraform_init

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/http...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/http v3.5.0...
- Installed hashicorp/http v3.5.0 (signed by HashiCorp)
- Installing hashicorp/aws v6.27.0...
- Installed hashicorp/aws v6.27.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.
```

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

```
@23-22411-061-rgb ~ ~/Lab12 $ |
```

task1_terraform_apply

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/http from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/http v3.5.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
@23-22411-061-rgb ~ ~/Lab12 $ aws configure
AWS Access Key ID [None]: A[REDACTED]C[REDACTED]B[REDACTED]
AWS Secret Access Key [None]: W[REDACTED]m[REDACTED]n[REDACTED]d[REDACTED]C[REDACTED]
Default region name [None]: me-central-1
Default output format [None]: json
@23-22411-061-rgb ~ ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]

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

Terraform will perform the following actions:

# aws_default_route_table.main_rt will be created
+ resource "aws_default_route_table" "main_rt" {
    + arn
    + default_route_table_id
    + id
    + owner_id
    + region
    + route
    + {
        + cidr_block
    }
}
```

task1_terraform_output

```
aws_instance_public_ip = "3.29.30.124"
@23-22411-061-rgb ~ ~/Lab12 $ terraform output
aws_instance_public_ip = "3.29.30.124"
@23-22411-061-rgb ~ ~/Lab12 $ |
```

task1_nginx_browser

Not secure 3.29.30.124

Welcome to nginx!

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

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

Thank you for using nginx.

task1_terraform_destroy

```
# [root@iZc1t-061 ~]# terraform destroy
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
aws_key_pair.ssh-key: Refreshing state... [id=serverkey]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-01ae138f9bd5ae563]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-0ecf3fa34f6ffa401]
aws_default_security_group.default_sg: Refreshing state... [id=sg-0dcdeaa58cf17342]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0787b495ce5230dc]
aws_default_route_table.main_rt: Refreshing state... [id=rtb-09e76bdf888ce9152]
aws_instance.myapp_server: Refreshing state... [id=i-0abfe53612f9b3a06]
```

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

- destroy

Terraform will perform the following actions:

```
# aws_default_route_table.main_rt will be destroyed
- resource "aws_default_route_table" "main_rt" {
  - arn          = "arn:aws:ec2:me-central-1:075006647027:route-table/rtb-09e76bdf888ce9152" -> null
  - default_route_table_id = "rtb-09e76bdf888ce9152" -> null
  - id          = "rtb-09e76bdf888ce9152" -> null
  - owner_id    = "075006647027" -> null
  - propagating_vgws = [] -> null
  - region      = "me-central-1" -> null
  - route       = [
      - {
          - cidr_block        = "0.0.0.0/0"
          - gateway_id        = "igw-0ecf3fa34f6ffa401"
          # (10 unchanged attributes hidden)
      },
    ] -> null
  - tags          = {
      - "Name" = "dev-rt"
    } -> null
  - tags_all     = {
      - "Name" = "dev-rt"
    } -> null
  - vpc_id        = "vpc-01ae138f9bd5ae563" -> null
}
```

```
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e
aws_instance.myapp_server: Still destroying... [id=i-0abfe5361
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e
aws_instance.myapp_server: Still destroying... [id=i-0abfe5361
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e
aws_instance.myapp_server: Still destroying... [id=i-0abfe5361
aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e
aws_internet_gateway.myapp_igw: Destruction complete after 48s
aws_instance.myapp_server: Still destroying... [id=i-0abfe5361
aws_instance.myapp_server: Still destroying... [id=i-0abfe5361
aws_instance.myapp_server: Destruction complete after 1m1s
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_subnet.myapp_subnet_1: Destroying... [id=subnet-0787b495ce
aws_default_security_group.default_sg: Destroying... [id=sg-0d
aws_default_security_group.default_sg: Destruction complete af
aws_key_pair.ssh-key: Destruction complete after 0s
aws_subnet.myapp_subnet_1: Destruction complete after 0s
aws_vpc.myapp_vpc: Destroying... [id=vpc-01ae138f9bd5ae563]
aws_vpc.myapp_vpc: Destruction complete after 1s
```

Destroy complete! Resources: 7 destroyed.

@23-22411-061-rgb ~ /Lab12 \$ |

Task 2 — Use remote-exec provisioner

task2_main_tf_remote_exec

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

resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6" # Amazon Linux 2023 Kernel 6.1 AMI
  instance_type = var.instance_type
  subnet_id     = aws_subnet.myapp_subnet_1.id
  security_groups = [aws_default_security_group.default_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

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

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

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

task2_terraform_apply

```
@23-22411-061-rgb ~ /Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]

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

Terraform will perform the following actions:

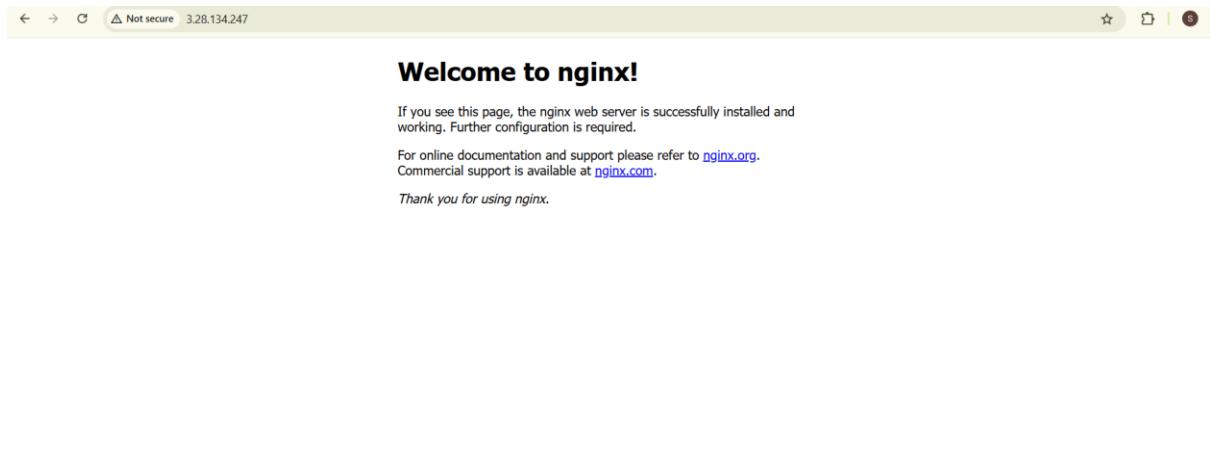
# aws_default_route_table.main_rt will be created
+ resource "aws_default_route_table" "main_rt" {
  + arn          = (known after apply)
  + default_route_table_id = (known after apply)
  + id           = (known after apply)
  + owner_id     = (known after apply)
  + region       = "me-central-1"
  + route        = [
    + {
      + cidr_block      = "0.0.0.0/0"
      + gateway_id     = (known after apply)
      + # (10 unchanged attributes hidden)
    },
  ],
  + tags          = {
    + "Name" = "dev-rt"
  }
  + tags_all      = {
    + "Name" = "dev-rt"
  }
  + vpc_id        = (known after apply)
}

# aws_default_security_group.default_sg will be created
+ resource "aws_default_security_group" "default_sg" {
  + arn          = (known after apply)
  + description   = (known after apply)
  + egress        = [
    + {
      + from_port    = 0
      + to_port     = 0
      + protocol    = "tcp"
      + rule_name   = "HTTP"
      + security_group_id = (known after apply)
      + type        = "ingress"
    },
  ],
  + ingress        = [
    + {
      + from_port    = 0
      + to_port     = 0
      + protocol    = "tcp"
      + rule_name   = "HTTP"
      + security_group_id = (known after apply)
      + type        = "egress"
    },
  ],
  + name          = "default_sg"
  + owner_id     = (known after apply)
  + rules         = [
    + {
      + from_port    = 0
      + to_port     = 0
      + protocol    = "tcp"
      + rule_name   = "HTTP"
      + security_group_id = (known after apply)
      + type        = "ingress"
    },
  ],
  + tags          = {
    + "Name" = "default_sg"
  }
}
```

task2_terraform_output

```
@23-22411-061-rgb ~ /Lab12 $ terraform output
aws_instance_public_ip = "3.28.134.247"
```

task2_nginx_browser



Task 3 — Use file and local-exec provisioners

task3_main_tf_all_provisioners

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

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

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

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

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

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

:wq!
```

task3_terraform_apply

```

023-22411-061-rgb ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]

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

Terraform will perform the following actions:

# aws_default_route_table.main_rt will be created
+ resource "aws_default_route_table" "main_rt" {
    + arn
    + default_route_table_id = (known after apply)
    + id
    + owner_id
    + region
    + route
        + [
            + {
                + cidr_block
                + gateway_id
                + # (10 unchanged attributes hidden)
            },
        ]
    + tags
        + "Name" = "dev-rt"
    + tags_all
        + "Name" = "dev-rt"
    + vpc_id
}

# aws_default_security_group.default_sg will be created
+ resource "aws_default_security_group" "default_sg" {
    + arn
    + description
    + egress
        + [
            + cidr_blocks
                + "0.0.0.0/0",
        ],
}

```

task3_terraform_output

```

aws_instance_public_ip = "51.112.43.42"
@23-22411-061-rgb ~/Lab12 $ terraform output
aws_instance_public_ip = "51.112.43.42"
@23-22411-061-rgb ~/Lab12 $ ...

```

task3_nginx_browser



task3_terraform_destroy

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform destroy
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
aws_key_pair.ssh-key: Refreshing state... [id=serverkey]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-02380df936b30ec9f]
aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-05767ff26504d4e8a]
aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-0731d26df8c66e383]
aws_default_security_group.default_sg: Refreshing state... [id=sg-0e084a2ccb0218059]
aws_default_route_table.main_rt: Refreshing state... [id=rtb-022dd93098ba450eb]
aws_instance.myapp-server: Refreshing state... [id=i-0142b433b5a110470]

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

Terraform will perform the following actions:

# aws_default_route_table.main_rt will be destroyed
- resource "aws_default_route_table" "main_rt" {
  - arn = "arn:aws:ec2:me-central-1:075006647027:route-table/rtb-022dd93098ba450eb" -> null
  - default_route_table_id = "rtb-022dd93098ba450eb" -> null
  - id = "075006647027" -> null
  - owner_id = []
  - propagating_vgws = []
  - region = "me-central-1" -> null
  - route = [
    - {
      - cidr_block = "0.0.0.0/0"
      - gateway_id = "igw-05767ff26504d4e8a"
      - # (10 unchanged attributes hidden)
    },
    ] -> null
  - tags = {
    - "Name" = "dev-rt"
    } -> null
  - tags_all = {
    - "Name" = "dev-rt"
    } -> null
  - vpc_id = "vpc-02380df936b30ec9f" -> null
}
```

task3_main_tf_restored

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

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

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

Task 4 — Create Terraform modules (subnet module)

task4_module_structure

```
@23-22411-061-rgb ~ ~/Lab12 $ vim main.tf
@23-22411-061-rgb ~ ~/Lab12 $ mkdir -p modules/subnet
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/subnet/main.tf
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/subnet/variables.tf
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/subnet/outputs.tf
```

task4_subnet_variables

```
codespace@codespaces-6aecc8: vim modules/subnet/variables.tf
variable "vpc_id" {}
variable "subnet_cidr_block" {}
variable "availability_zone" {}
variable "env_prefix" {}
variable "default_route_table_id" {}

~
```

task4_subnet_main

```
variable "vpc_id" {}
    } -> null
resource "aws_subnet" "myapp_subnet_1" {
  vpc_id      = var.vpc_id
  cidr_block = var.subnet_cidr_block
  availability_zone = var.availability_zone
  map_public_ip_on_launch = true
  tags = {
    Name = "${var.env_prefix}-subnet-1"
  }
}

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

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.myapp_igw.id
  }
  tags = {
    Name = "${var.env_prefix}-rt"
  }
}

resource "aws_internet_gateway" "myapp_igw" {
  vpc_id = var.vpc_id
  tags = {
    Name = "${var.env_prefix}-igw"
  }
}
```

task4_subnet_outputs

```
output "subnet" {
  value = aws_subnet.myapp_subnet_1
}
```

task4_main_tf_with_module

```
protocol      = "tcp"
  cidr_blocks = ["0.0.0.0/0"]
}
egress {
  from_port    = 0
  to_port      = 0
  protocol     = "-1"
  cidr_blocks  = ["0.0.0.0/0"]
  prefix_list_ids = []
}
tags = {
  Name = "${var.env_prefix}-default-sg"
}
}

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

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

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

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

task4_terraform_init

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-subnet in modules/subnet
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.
@23-22411-061-rgb ~ ~/Lab12 $ |
```

task4_terraform_apply

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]

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

Terraform will perform the following actions:

# aws_default_security_group.default_sg will be created
+ resource "aws_default_security_group" "default_sg" {
    + arn           = (known after apply)
    + description   = (known after apply)
    + egress        = [
        + {
            + cidr_blocks = [
                + "0.0.0.0/0",
            ]
            + from_port   = 0
            + ipv6_cidr_blocks = []
            + prefix_list_ids = []
            + protocol    = "tcp"
            + security_groups = []
            + self         = false
            + to_port     = 0
            # (1 unchanged attribute hidden)
        },
    ]
    + id           = (known after apply)
    + ingress      = [
        + {
            + cidr_blocks = [
                + "0.0.0.0/0",
            ]
            + from_port   = 80
            + ipv6_cidr_blocks = []
            + prefix_list_ids = []
            + protocol    = "tcp"
            + security_groups = []
            + self         = false
        }
    ]
}
```

task4_terraform_output

```
aws_instance_public_ip = "3.28.46.250"
@23-22411-061-rgb ~ ~/Lab12 $ terraform output
aws_instance_public_ip = "3.28.46.250"
@23-22411-061-rgb ~ ~/Lab12 $ |
```

task4_nginx_browser

The screenshot shows a web browser window with the address bar containing 'Not secure 3.28.46.250'. The main content is the 'Welcome to nginx!' page, which includes a message about successful installation, links to online documentation and commercial support, and a thank you note.

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.
For online documentation and support please refer to [nginx.org](#).
Commercial support is available at [nginx.com](#).
Thank you for using nginx.

Task 5 — Create webserver module

task5_webserver_module_structure

```
@23-22411-061-rgb ~ ~/Lab12 $ mkdir -p modules/webserver
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/webserver/main.tf
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/webserver/variables.tf
@23-22411-061-rgb ~ ~/Lab12 $ touch modules/webserver/outputs.tf
@23-22411-061-rgb ~ ~/Lab12 $
```

task5_webserver_variables

```
variable "env_prefix" {}
variable "instance_type" {}
variable "availability_zone" {}
variable "public_key" {}
variable "my_ip" {}
variable "vpc_id" {}
variable "subnet_id" {}
variable "script_path" {}
variable "instance_suffix" []
```

```
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~  
~
```

task5_webserver_main

```

    cidr_blocks = ["0.0.0.0/0"]
}
ingress {
  from_port   = 80
  to_port     = 80
  protocol   = "tcp"
  cidr_blocks = ["0.0.0.0/0"]
}
egress {
  from_port   = 0
  to_port     = 0
  protocol   = "-1"
  cidr_blocks = ["0.0.0.0/0"]
  prefix_list_ids = []
}
tags = {
  Name = "${var.env_prefix}-default-sg"
}
}

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

resource "aws_instance" "myapp-server" {
  ami           = "ami-05524d6658fcf35b6" # Amazon Linux 2023 Kernel 6.1 AMI
  instance_type = var.instance_type
  subnet_id     = var.subnet_id
  vpc_security_group_ids = [aws_security_group.web_sg.id]
  availability_zone = var.availability_zone
  associate_public_ip_address = true
  key_name = aws_key_pair.ssh-key.key_name

  user_data = file(var.script_path)

  tags = {
    Name = "${var.env_prefix}-ec2-instance-${var.instance_suffix}"
  }
}
:wq!

```

task5_webserver_outputs

```

<----> https://www.terraform.io/docs/language/functions/output.html
output "aws_instance" {
  value = aws_instance.myapp-server
}
~
~
```

task5_main_tf_webserver_module

```
module "myapp-subnet" {
  source = "./modules/subnet"
  vpc_id = aws_vpc.myapp_vpc.id
  subnet_cidr_block = var.subnet_cidr_block
  availability_zone = var.availability_zone
  env_prefix = var.env_prefix
  default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id
}

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

task5_outputs_updated

```
└── outputs.tf
output "webserver_public_ip" {
  value = module.myapp-webserver.aws_instance.public_ip
}
```

task5_terraform_init

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-webserver in modules/webserver
Initializing provider plugins...
- Reusing previous version of hashicorp/http from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/http v3.5.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
@23-22411-061-rgb ~ ~/Lab12 $ |
```

task5_terraform_apply

```

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

Changes to Outputs:
  - aws_instance_public_ip = "3.28.46.250" -> null
  + webserver_public_ip      = (known after apply)
aws_instance.myapp-server: Destroying... [id=i-01076fcfb63bd674]
module.myapp-webserver.aws_key_pair.ssh-key: Creating...
module.myapp-webserver.aws_security_group.web_sg: Creating...
module.myapp-webserver.aws_key_pair.ssh-key: Creation complete after 0s [id=dev-serverkey-0]
module.myapp-webserver.aws_security_group.web_sg: Creation complete after 3s [id=sg-007ea724f1fee40d6]
module.myapp-webserver.aws_instance.myapp-server: Creating...
aws_instance.myapp-server: Still destroying... [id=i-01076fcfb63bd674, 00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m10s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-01076fcfb63bd674, 00m20s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m20s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-01076fcfb63bd674, 00m30s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m30s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-01076fcfb63bd674, 00m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m40s elapsed]
aws_instance.myapp-server: Still destroying... [id=i-01076fcfb63bd674, 00m50s elapsed]
aws_instance.myapp-server: Destruction complete after 52s
aws_key_pair.ssh-key: Destroying... [id=serverkey]
aws_default_security_group.default_sg: Destroying... [id=sg-075f2f182352479ee]
aws_default_security_group.default_sg: Destruction complete after 0s
aws_key_pair.ssh-key: Destruction complete after 0s
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m50s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m00s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m20s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m30s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [01m50s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m00s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m20s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m30s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [02m50s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still creating... [03m00s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Creation complete after 3m9s [id=i-03f51baec85492b92]

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

Outputs:

webserver_public_ip = "3.28.215.203"
@23-22411-061-rgb ~ ~/Lab12 $ |

```

task5_terraform_output

```

webserver_public_ip = "3.28.215.203"
@23-22411-061-rgb ~ ~/Lab12 $ terraform output
webserver_public_ip = "3.28.215.203"
@23-22411-061-rgb ~ ~/Lab12 $ |

```

task5_nginx_browser

← → ⌂ △ Not secure 3.28.215.203

Welcome to nginx!

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

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

Thank you for using nginx.

task5_terraform_destroy

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

Changes to Outputs:
  - webserver_public_ip = "3.28.215.203" -> null

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

Enter a value: yes

module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-0456bf57b94a7eb1c]
module.myapp-webserver.aws_instance.myapp-server: Destroying... [id=i-03f51baec85492b92]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id=igw-09b2b33bce3c7c075]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-03f51baec85492b92, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-09b2b33bce3c7c075, 00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-03f51baec85492b92, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-09b2b33bce3c7c075, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-09b2b33bce3c7c075, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-09b2b33bce3c7c075, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 37s
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-03f51baec85492b92, 00m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Destruction complete after 40s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id=subnet-0463313834d17431e]
module.myapp-webserver.aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-0]
module.myapp-webserver.aws_security_group.web_sg: Destroying... [id=sg-007ea724f1fee40d6]
module.myapp-webserver.aws_key_pair.ssh-key: Destruction complete after 0s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 1s
module.myapp-webserver.aws_security_group.web_sg: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-0d9b65b00400090d]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@23-22411-061-rgb ~ ~/Lab12 $ |
```

Task 6 — Configure HTTPS with self-signed certificates

task6_entry_script_https

```
http {
    log_format main '$remote_addr - $remote_user [$time_local] "$request"
                    '$status '$body_bytes_sent "$http_referer"
                    '"$http_user_agent" "$http_x_forwarded_for"';
    access_log /var/log/nginx/access.log main;

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

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

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

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

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

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

systemctl restart nginx
~
```

task6_terraform_apply

```

@23-22411-061-rgb ~ ~/Lab12 $ vim entry-script.sh
@23-22411-061-rgb ~ ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]

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

Terraform will perform the following actions:

# aws_vpc.myapp_vpc will be created
+ resource "aws_vpc" "myapp_vpc" {
  + arn = (known after apply)
  + cidr_block = "10.0.0.0/16"
  + default_network_acl_id = (known after apply)
  + default_route_table_id = (known after apply)
  + default_security_group_id = (known after apply)
  + dhcp_options_id = (known after apply)
  + enable_dns_hostnames = (known after apply)
  + enable_dns_support = true
  + enable_network_address_usage_metrics = (known after apply)
  + id = (known after apply)
  + instance_tenancy = "default"
  + ipv6_association_id = (known after apply)
  + ipv6_cidr_block = (known after apply)
  + main_route_table_id = (known after apply)
  + owner_id = (known after apply)
  + region = "me-central-1"
  + tags = {
    + "Name" = "dev-vpc"
  }
  + tags_all = {
    + "Name" = "dev-vpc"
  }
}

# module.myapp-subnet.aws_default_route_table.main_rt will be created
+ resource "aws_default_route_table" "main_rt" {
  + arn = (known after apply)
  + default_route_table_id = (known after apply)

    + security_groups = []
    + self = false
    + to_port = 22
    # (1 unchanged attribute hidden)
  ],
  + name = "dev-web-sg-0"
  + name_prefix = (known after apply)
  + owner_id = (known after apply)
  + region = "me-central-1"
  + revoke_rules_on_delete = false
  + tags = {
    + "Name" = "dev-default-sg"
  }
  + tags_all = {
    + "Name" = "dev-default-sg"
  }
  + vpc_id = (known after apply)
}

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

Changes to Outputs:
+ webserver_public_ip = (known after apply)
module.myapp-webserver.aws_key_pair.ssh-key: Creating...
aws_vpc.myapp_vpc: Creating...
module.myapp-webserver.aws_key_pair.ssh-key: Creation complete after 0s [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Creation complete after 1s [id=vpc-0410b9b8db6f8d9db]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creating...
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creating...
module.myapp-webserver.aws_security_group.web_sg: Creating...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creation complete after 1s [id=igw-07a753edb12402b67]
module.myapp-subnet.aws_default_route_table.main_rt: Creating...
module.myapp-subnet.aws_default_route_table.main_rt: Creation complete after 1s [id=rtb-07ecf971dd3af53f6]
module.myapp-webserver.aws_security_group.web_sg: Creation complete after 3s [id=sg-0f97f2e6feb675cf]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Still creating... [00m10s elapsed]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creation complete after 11s [id=subnet-05532ac0c39950c21]
module.myapp-webserver.aws_instance.myapp_server: Creating...
module.myapp-webserver.aws_instance.myapp_server: Still creating... [00m10s elapsed]
module.myapp-webserver.aws_instance.myapp_server: Creation complete after 12s [id=i-0a307ad827fd0ffd8]

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

Outputs:
webserver_public_ip = "51.112.228.197"
@23-22411-061-rgb ~ ~/Lab12 $ |

```

task6_terraform_output

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

Outputs:

```

webserver_public_ip = "51.112.228.197"
@23-22411-061-rgb ~ ~/Lab12 $ terraform output
webserver_public_ip = "51.112.228.197"
@23-22411-061-rgb ~ ~/Lab12 $ |

```

task6_browser_security_warning

The screenshot shows a browser window with the URL <https://51.112.228.197>. A red warning bar at the top says "Not secure". Below it, a large red exclamation mark icon is displayed. The main content area has the following text:

Your connection is not private
Attackers might be trying to steal your information from **51.112.228.197** (for example, passwords, messages, or credit cards). [Learn more about this warning](#)

NET::ERR_CERT_AUTHORITY_INVALID

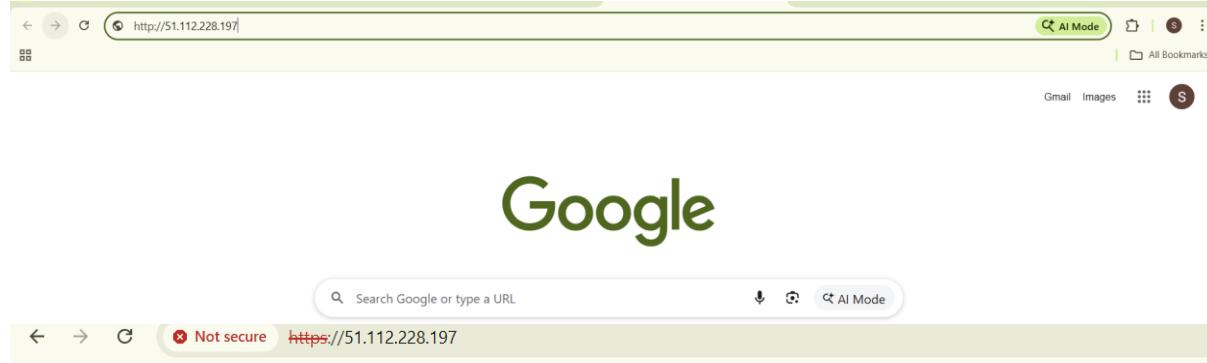
A button with a question mark icon and the text "Turn on enhanced protection to get Chrome's highest level of security" is visible. At the bottom, there are "Advanced" and "Back to safety" buttons.

task6_nginx_https_browser

The screenshot shows a browser window with the URL <https://51.112.228.197>. A red warning bar at the top says "Not secure". The main content area displays the "Welcome to nginx!" page, which includes the following text:

Welcome to nginx!
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.
For online documentation and support please refer to [nginx.org](#).
Commercial support is available at [nginx.com](#).
Thank you for using nginx.

task6_http_redirect



Welcome to nginx!

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

For online documentation and support please refer to [nginx.org](#). Commercial support is available at [nginx.com](#).

Thank you for using nginx.

Task 7 — Configure Nginx as reverse proxy

task7_apache_script

```
#!/bin/bash
yum update -y
yum install httpd -y
systemctl start httpd
systemctl enable httpd
echo "<h1>Welcome to My Web Server</h1>" > /var/www/html/index.html
hostnamectl set-hostname myapp-websrvr
echo "<h2>Hostname: ${hostname}</h2>" >> /var/www/html/index.html
TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" \
-H "X-aws-ec2-metadata-token-ttl-seconds: 21600")
echo "<h2>Private IP: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/local-ipv4)</h2>" >> /var/www/html/index.html
echo "<h2>Public IP: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-ipv4)</h2>" >> /var/www/html/index.html
echo "<h2>Public DNS: $(curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-hostname)</h2>" >> /var/www/html/index.html
echo "<h2>Deployed Via Terraform</h2>" >> /var/www/html/index.html
```

task7_main_tf_web1

```

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

module "myapp-subnet" {
  source = "./modules/subnet"
  vpc_id = aws_vpc.myapp_vpc.id
  subnet_cidr_block = var.subnet_cidr_block
  availability_zone = var.availability_zone
  env_prefix = var.env_prefix
  default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id
}

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

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

```

task7_outputs_web1

```

codespace@codespaces-6aec8: vim outputs.tf
output "webserver_public_ip" {
  value = module.myapp-webserver.aws_instance.public_ip
}

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

task7_terraform_apply

```
codepace@codespaces-6acc2: terraform apply -auto-approve
023-22411-061-rgb ~~/Lab12 $ vim apache.sh
023-22411-061-rgb ~~/Lab12 $ vim main.tf
023-22411-061-rgb ~~/Lab12 $ vim outputs.tf
023-22411-061-rgb ~~/Lab12 $ terraform apply -auto-approve
| Error: Module not installed
|   on main.tf line 35:
|     35: module "myapp-web-1" {
|
| This module is not yet installed. Run "terraform init" to install all modules required by this configuration.
023-22411-061-rgb ~~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-web-1 in modules/webserver
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/http from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/http v3.5.0
Terraform has been successfully initialized!
You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.
If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
023-22411-061-rgb ~~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
module.myapp-webservice.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-0410b9b8db6f8d9db]
module.myapp-webservice.aws_security_group.web_sg: Refreshing state... [id=sg-0ff97f2e6feb675cf]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-07a753edb12402b67]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-05532ac0c39950c21]
module.myapp-webservice.aws_instance.myapp_server: Refreshing state... [id=i-0a307ad827fd0ffd8]
module.myapp-subnet.aws_default_route_table.main_rt: Refreshing state... [id=rtb-07ecf971dd3af53f6]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
+ create
- update in-place
-/+ destroy and then create replacement

Terraform will perform the following actions:
```

```

      name          = "dev-web-sg"
      tags          = {
        "Name" = "dev-web-sg-0"
      }
    # (9 unchanged attributes hidden)
}

Plan: 4 to add, 1 to change, 1 to destroy.

Changes to Outputs:
+ aws_web-1_public_ip = (known after apply)
- webserver_public_ip = '51.112.228.197' -> null
module.myapp-web-1.aws_key_pair.ssh-key: Creating...
module.myapp-webserver.aws_instance.myapp-server: Destroying... [id=i-0a307ad827fd0ffd8]
module.myapp-web-1.aws_security_group.web_sg: Creating...
module.myapp-web-1.aws_key_pair.ssh-key: Creation complete after 0s [id=dev-serverkey-1]
module.myapp-web-1.aws_security_group.web_sg: Creation complete after 3s [id=sg-0f56c1b91ea052f64]
module.myapp-1.aws_instance.myapp-server: Creating...
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a307ad827fd0ffd8, 00m10s elapsed]
module.myapp-1.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-web-1.aws_instance.myapp-server: Creation complete after 12s [id=i-0b44a5df236217f77]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a307ad827fd0ffd8, 00m20s elapsed]
module.myapp-1.aws_instance.myapp-server: Still destroying... [id=i-0a307ad827fd0ffd8, 00m30s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a307ad827fd0ffd8, 00m40s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a307ad827fd0ffd8, 00m50s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Destruction complete after 50s
module.myapp-webserver.aws_security_group.web_sg: Modifying... [id=sg-0f97f72e6feb675cf]
module.myapp-webserver.aws_security_group.web_sg: Modifications complete after 1s [id=sg-0f97f72e6feb675cf]
module.myapp-webserver.aws_instance.myapp-server: Creating...
module.myapp-webserver.aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver.aws_instance.myapp-server: Creation complete after 13s [id=i-031924a29cf02bc97]

Apply complete! Resources: 4 added, 1 changed, 1 destroyed.

Outputs:

aws_web-1_public_ip = "51.112.167.85"
@23-22411-061-rgb → ~/Lab12 $ |
  _/m'-'_
[ec2-user@myapp-webserver ~]$ sudo vim /etc/nginx/nginx.conf
[ec2-user@myapp-webserver ~]$ ls -l /etc/nginx/
ls: cannot access '/etc/nginx/': No such file or directory
[ec2-user@myapp-webserver ~]$ hostname
myapp-webserver
[ec2-user@myapp-webserver ~]$ exit
Logout
Connection to 51.112.167.85 closed.
@23-22411-061-rgb → ~/Lab12 $ vim outputs.tf
@23-22411-061-rgb → ~/Lab12 $ terraform output
aws_web-1_public_ip = "51.112.167.85"
@23-22411-061-rgb → ~/Lab12 $ vim outputs.tf
@23-22411-061-rgb → ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
module.myapp-web-1.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-1]
module.myapp-webserver.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-0410b98db6f8d9db]
module.myapp_subnet.aws_internet_gateway.myapp_iow: Refreshing state... [id=iow-07a753edb12402b67]
module.myapp-web-1.aws_security_group.web_sg: Refreshing state... [id=sg-0f56c1b91ea052f64]
module.myapp_subnet.aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-05532ac0c39950c21]
module.myapp-webserver.aws_security_group.web_sg: Refreshing state... [id=sg-0f97f72e6feb675cf]
module.myapp_subnet.aws_default_route_table.main_rt: Refreshing state... [id=rtb-07ecf971dd3af53f6]
module.myapp-1.aws_instance.myapp-server: Refreshing state... [id=i-0b44a5df236217f77]
module.myapp-webserver.aws_instance.myapp-server: Refreshing state... [id=i-031924a29cf02bc97]

Changes to Outputs:
+ webserver_public_ip = "3.29.125.33"

You can apply this plan to save these new output values to the Terraform state, without changing any real infrastructure.

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

Outputs:

aws_web-1_public_ip = "51.112.167.85"
webserver_public_ip = "3.29.125.33"

```

task7_terraform_output

```

@23-22411-061-rgb → ~/Lab12 $ terraform output
aws_web-1_public_ip = "51.112.167.85"
webserver_public_ip = "3.29.125.33"
@23-22411-061-rgb → ~/Lab12 $ |

```

task7_ssh_webserver

```

aws_web_1_public_ip = "51.112.167.85"
webserver_public_ip = "3.29.125.33"
@23-22411-061-rgb ~ ~/Lab12 $ vim outputs.tf
@23-22411-061-rgb ~ ~/Lab12 $ ssh ec2-user@3.29.125.33
The authenticity of host '3.29.125.33 (3.29.125.33)' can't be established.
ED25519 key fingerprint is SHA256:YVtaaifKA4/Hg3WQMqTckBVT8OrJUoJ78085DzkmIQk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.29.125.33' (ED25519) to the list of known hosts.

      #_
      ~\_\####_          Amazon Linux 2023
      ~~\_\####\_
      ~~ \###]
      ~~  \#/ ___
      ~~   V~,' '-'> https://aws.amazon.com/linux/amazon-linux-2023
      ~~
      ~~ .-' / \
      ~~ /-' / \
      _/m/ .-' /
[ec2-user@ip-10-0-10-51 ~]$ |

```

task7_nginx_conf_reverse_proxy

```

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

server {
    listen 443 ssl;
    server_name 3.29.125.33;
    ssl_certificate /etc/ssl/certs/selfsigned.crt;
    ssl_certificate_key /etc/ssl/private/selfsigned.key;

    location / {
        root /usr/share/nginx/html;
        index index.html;
        # proxy_pass http://51.112.167.85:80;
        # proxy_pass http://backend_servers;
    }
}

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

```

task7_nginx_restart

```

[ec2-user@ip-10-0-10-51 ~]$ sudo vim /etc/nginx/nginx.conf
[ec2-user@ip-10-0-10-51 ~]$ sudo systemctl restart nginx

```

task7_error_log

```

[ec2-user@ip-10-0-10-51 ~]$ cat /var/log/nginx/error.log
2026/01/06 03:54:41 [notice] 2933#2933: using the "epoll" event method
2026/01/06 03:54:41 [notice] 2933#2933: nginx/1.28.0
2026/01/06 03:54:41 [notice] 2933#2933: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2026/01/06 03:54:41 [notice] 2933#2933: getrlimit(RLIMIT_NOFILE): 65535:65535
2026/01/06 03:54:41 [notice] 2977#2977: start worker processes
2026/01/06 03:54:41 [notice] 2977#2977: start worker process 2980
2026/01/06 03:54:41 [notice] 2977#2977: start worker process 2981
2026/01/06 03:54:41 [notice] 2977#2977: signal 9 (SIGQUIT) received from 1, shutting down
2026/01/06 03:54:41 [notice] 2980#2980: gracefully shutting down
2026/01/06 03:54:41 [notice] 2980#2980: exiting
2026/01/06 03:54:41 [notice] 2980#2980: exit
2026/01/06 03:54:41 [notice] 2981#2981: gracefully shutting down
2026/01/06 03:54:41 [notice] 2981#2981: exiting
2026/01/06 03:54:41 [notice] 2981#2981: exit
2026/01/06 03:54:41 [notice] 2977#2977: signal 17 (SIGCHLD) received from 2981
2026/01/06 03:54:41 [notice] 2977#2977: worker process 2981 exited with code 0
2026/01/06 03:54:41 [error] 2977#2977: worker process 2981 terminated
2026/01/06 03:54:41 [notice] 2977#2977: signal 17 (SIGCHLD) received from 2980
2026/01/06 03:54:41 [notice] 2977#2977: worker process 2980 exited with code 0
2026/01/06 03:54:41 [notice] 2977#2977: exit
2026/01/06 03:54:41 [emerg] 3574#3574: invalid number of arguments in "pid" directive in /etc/nginx/nginx.conf:4
2026/01/06 04:09:27 [emerg] 25640#25640: invalid number of arguments in "pid" directive in /etc/nginx/nginx.conf:4
2026/01/06 04:10:12 [emerg] 25648#25648: invalid number of arguments in "pid" directive in /etc/nginx/nginx.conf:4
2026/01/06 04:10:52 [emerg] 25659#25659: invalid number of arguments in "pid" directive in /etc/nginx/nginx.conf:4
2026/01/06 04:11:20 [notice] 25723#25723: using the "epoll" event method
2026/01/06 04:11:20 [notice] 25723#25723: OS: Linux 6.1.158-180.294.amzn2023.x86_64
2026/01/06 04:11:20 [notice] 25723#25723: getrlimit(RLIMIT_NOFILE): 65535:65535
2026/01/06 04:11:20 [notice] 25724#25724: start worker processes
2026/01/06 04:11:20 [notice] 25724#25724: start worker process 25725
2026/01/06 04:11:20 [notice] 25724#25724: start worker process 25726
2026/01/06 04:15:59 [error] 25726#25726: "open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 154.192.16.58, server: 3.29.125.33, request: "GET /favicon.ico HTTP/1.1", host: "3.29.125.33", referer: "https://3.29.125.33/"
```

task7_access_log

```
2026/01/06 04:15:59 [error] 25726#25726: *8 open() "/usr/share/nginx/html/favicon.ico" failed (2: No such file or directory), client: 154.192.16.58, server: 3.29.125.33, reuest: "GET /favicon.ico HTTP/1.1", host: "3.29.125.33", referrer: "https://3.29.125.33/"  
[ec2-user@ip-10-0-10-51 ~]$ cat /var/log/nginx/access.log  
154.192.16.58 - - [06/Jan/2026:04:15:59 +0000] "GET / HTTP/1.1"200 615 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
154.192.16.58 - - [06/Jan/2026:04:15:59 +0000] "GET /favicon.ico HTTP/1.1"404 555 "https://3.29.125.33/Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
154.192.16.58 - - [06/Jan/2026:04:15:59 +0000] "GET / HTTP/1.1"200 615 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
154.192.16.58 - - [06/Jan/2026:04:19:48 +0000] "GET / HTTP/1.1"200 615 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
154.192.16.58 - - [06/Jan/2026:04:19:49 +0000] "GET / HTTP/1.1"200 615 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
154.192.16.58 - - [06/Jan/2026:04:19:52 +0000] "GET / HTTP/1.1"200 615 "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36"  
[ec2-user@ip-10-0-10-51 ~]$
```

task7_mime_types

```
[root@i-0000000000000000 ~]$ cat /etc/nginx/mime.types  
types {  
application/A2L a2l;  
application/AML aml;  
application/andrew-inset ez;  
application/ATF atf;  
application/ATFX atfx;  
application/ATXML atxml;  
application/atom+xml atom;  
application/atomcat+xml atomcat;  
application/atomdeleted+xml atomdeleted;  
application/atomsvc+xml atomsvc;  
application/atsc-dwd+xml dwd;  
application/atsc-held+xml held;  
application/atsc-rsat+xml rsat;  
application/auth-policy+xml apxml;  
application/bacnet-xdd+zip xdd;  
application/calendar+xml xcs;  
application/cbor cbor;  
application/cccex c3ex;  
application/ccmp+xml ccmp;  
application/ccxml+xml ccxml;  
application/CDFX+XML cdfx;  
application/cdmi-capability cdmia;  
application/cdmi-container cdmic;  
application/cdmi-domain cdmid;  
application/cdmi-object cdmio;  
application/cdmi-queue cdmiq;  
application/CEA cea;  
application/cellml+xml cellml cml;  
application/clue_info+xml clue;  
application/cms cmst;  
application/cpl+xml cpl;  
application/csrattr csrattr;  
application/dash+xml mpd;  
application/dashdelta mpdd;  
application/davmount+xml davmount;  
application/DCD dcd;  
application/dicom dcm;  
application/DII dii;  
application/DIT dit;  
application/dskpp+xml xmls;  
application/dssc+der dssc;  
application/dssc+xml xdssc;  
application/dvcs dvc;  
application/ecmascript es;  
application/efi efi;
```

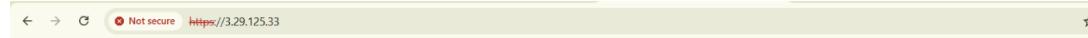
task7_ssl_cert

```
[ec2-user@ip-10-0-10-51 ~]$ cat /etc/ssl/certs/selfsigned.crt
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIUM5BP9CSztt7JXb04FX081lsHrcswDQYJKoZIhvcNAQEL
BQAwFjEUMBIGA1UEAwLMy4yOS4xMjUuMzMwHhcNMjYwMTA2MDM1NDQxWhcNMjcw
MTA2MDM1NDQxWjAwMRQwEgYDVQQDDAsLjI5LjEyNS4zMzCCASIwDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBALsxGpLk33S3DKMVpTu1IFvyNeq1P8RpB7RJn7mj
bJp9w0rZU3A2sPFJvSak1rbxXDvAUVmvpzsSgykzShb3VnCtuSBsvaETJAgY3xe
5FoNoEu2KETqCKGavr8VTU9H20IKNP510r1uFARffg3IZJ+AIjC6fYbUYUFNUxsR
honso8pZnjPbfA6yuju3VbaoydsWEdjb0jYZ9L0peM+nhU1TXxnTT1TfjejcyfQ6
/9qtLiYjBuHGSz7JDlf0vNFEI78yUPK7u8JnvxekvrA/BV2XqEqawZIB5biD+74
ckpBxvnrtQ9fy6x/rzq8tuwbb16bZd2Q071Qh5gdzw7hpdmCAwEAAaOBgDB+MB0G
A1UdDgQWBQm8UUakmbpVC8J6ASGSV13qEuJ2TAFBgNVHSMEGDAwgbQm8UUakmbp
VC8J6ASGSV13qEuJ2TAPBgNVHREECDAGhwQDHX0hMAkGA1UdEwQCMAwCwYDVR0P
BAQDAgWgMBMGA1UdJQQMMAoGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUAA4IBAQct
VUmRrlV/3I9wf010Qogd6FW6wX1FXUElpV3wc8rDpnw0z7znoVz1Mrq42vSB9J7
u55zMwtxddxmHpg2bs111ZCbNUVum8vmFwVvpfVV7B906waw/dsGTGEfv4BBiE1
ZQcmdQ6NXM5sqah3UPOKSU8vpzfP3RSO/DWKfQFr+1TukAdoSjdXnHRYhV7nIOkW
8h/hH2HCaxRZQnEAWJ/AjD8cwGYuSvhslmESRfsXg50DZisiegCW6AwVslo1k+JI
cwvAu6KkCD+1c0wDTwZ1BxNxptDa9qHw8ZnPJCrvcf9/ZyDSRhDWxm7WgwfaRIC
xJbdIAxX1Wpmo4D5hmSQ
-----END CERTIFICATE-----
[ec2-user@ip-10-0-10-51 ~]$ |
```

task7_ssl_key

```
[ec2-user@ip-10-0-10-51 ~]$ sudo cat /etc/ssl/private/selfsigned.key
-----BEGIN PRIVATE KEY-----
MIIEvwIBADANBgkqhki9w0BAQEFAASCBKkwggS1AgEAAoIBAAC7MRqS5N90twyj
FaU7tsBb8jXqpT/EaQeOSZ+5o2yafcNK2VNwNrDxSb0mpJa28Vw7wFFZr6c6bEoM
pMoow91ZwrbkgbL2hEyQIGN8XuRaDaBLtihE6gihr6/FU1PR9tCCjt+ddk9bhQE
X34NyGsfCIwun2G1GFHzVmBEYaJ7KPKWz4z23wOsro791w2qMnbFhHSW9I2gfSz
qXjPp4VNU18Z005U343o3Mn00v/ars4mIwbhxks+yQ2y39LzRRC0/M1Dyu7vcZ78
Xpl6wPwvd16hKmlmAew4g/u+HJKQcb5600PXusf686vLb1m29em2XdkDu5UIey
A81u4aXTAgMBAEAEcgEVjyY8XNFpNvX7IvcQ1V+Qt2i2San6eQ5GGWSoAMZ1VJ6
1dcpD9kiSwadjv/1zwICa8R9tc0mvtPsiUai/STVJLT/b0YTfF4Wp49ERdexe4sr
fbw9d4Iz19lizsV5iDyivh09r7y4Cwzc6+bSO6dGM/Tjy28exS6yY1HFJ4M7Z+5C
B7P09Y0Td3aa3gIerobr+4ouhH93IE7ApUN2PCgZMXqjVJ03I9MbWQ1VDM97Z1UD2
UKSh/RDj8Ya28KMnW7Da8bqOpIQ9oMvPyXgo/kdvnOxwwy6jHzPG5uJE55gpywu
Bd1Tw8eaG0XZceIUbCq277e9bKGmrYcOKIE88K18gQKBgQDyX+jjsqbTwGKx9jsu
9ZTr8bnUp9Hgm/TGP6cV6p3h40aEmTj3cagFOBFr0ibtecfbuDgR29r0SjVV0e+1
xPy6RH5Po/INOv9gdkdQy/Xtafo9nQc4mjHb2V8ux8+QZM6DoezKKrCwaLw3Zgv
eI1VVuQifVfZUYvfdNRW4W+JwKBgQDFtwpuZ219jFxm58Bqg9uPiCkQxfcwOCAN
NN3kqkW939Zr/jtKoiUXSjLC6/HtVojJq1eFnd1x5zQIT7tNs3s750H+1KQk2+JFe
YqOJuVz3raTaj08HCBVIw8NaPjm6Bi/hwS1expgFoufLRkDUWQSwF4FEYrXOE3G
EfTP1MySdQKBgQDOGwTrn8Xrt321+7ndNfTH1yd6S6NLrxM9Ko+5MwPE3PHwfCNX
OA07MuyN1Y2Iz9BGZac4eI7+wcQNEJMLkjGfxGM88hs5wApNCwemC2s70qzYIPGo
R1f3kCDrkQqtNDiwYRz+2wqbpLuaE2rwIXxFh61vYTgx11e705akJsbQKBgQCK
4fbjB36j4vAUKWT3uZqk4mRMb/B1KT0hh1seHgCCHRv2TjgoD3ybJdywLZA9+5Jrj
GaagsPqB4QPA18Td4Bn/n28zzs0CtFAQ0za0AOyka8PS8Tyujz1cQFD0ZKB3jMD
3RVxInEZejf3/4iFD180fBrssaq9vQivc8PnwXGZGQKBgQDXLNT47vM/n4cfQzc/
UHWg7gzesbg8vBpqYYEjF3FnUAbARnx1Nb9VQTwnQuBTHPxNnR9JRVVXqyL42CP/
Yj0+vX5wSnG0jV+MBgmwu09+p+9/05sfAd+Vn+0B2Cv/NdhJrpXKwtney0VjfMn6
sZdT+KLVVjnuZSCU6S43MVQi5A==
-----END PRIVATE KEY-----
[ec2-user@ip-10-0-10-51 ~]$ |
```

task7_reverse_proxy_browser



Task 8 — Configure Nginx as load balancer

task8_main_tf_web2

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

task8_outputs_web2

```
task8_outputs_web2
└── outputs.tf
    output "webserver_public_ip" {
        value = module.myapp-webserver.aws_instance.public_ip
    }

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

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

task8_terraform_apply

```
@23-22411-061-rgb ~/Lab12 $ vim outputs.tf
@23-22411-061-rgb ~/Lab12 $ terraform apply -auto-approve
Error: Module not installed
  on main.tf line 48:
  48: module "myapp-web-2" {

This module is not yet installed. Run "terraform init" to install all modules required by this configuration.

@23-22411-061-rgb ~/Lab12 $ terraform init
Initializing the backend...
Initializing modules...
- myapp-web-2 in modules/webserver
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/http from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/http v3.5.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
@23-22411-061-rgb ~/Lab12 $ terraform apply -auto-approve
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
module.myapp-webserver.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-0410b98db6f8d9db]
module.myapp-web-1.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-1]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-05532ac0c39950c21]
module.myapp-web-1.aws_security_group.web_sg: Refreshing state... [id=sg-0f56c1b91ea052f64]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-07a753edb12402b67]
module.myapp-webserver.aws_security_group.web_sg: Refreshing state... [id=sg-0f97f72e6feb675cf]
module.myapp-subnet.aws_default_route_table.main_rt: Refreshing state... [id=rtb-07ecf971dd3af53f6]
module.myapp-web-1.aws_instance.myapp-server: Refreshing state... [id=i-0b44a5df236217f77]
module.myapp-webserver.aws_instance.myapp-server: Refreshing state... [id=i-031924a29cf02bc97]
```

task8_terraform_output

```
@23-22411-061-rgb ~/Lab12 $ terraform output
aws_web-1_public_ip = "51.112.167.85"
aws_web-2_public_ip = "3.29.244.171"
webserver_public_ip = "3.29.125.33"
```

```
task8_nginx_conf_load_balancer
keepalive_timeout 65;
types_hash_max_size 4096;

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

upstream backend_servers {
    server 51.112.167.85:80;
    server 3.29.24.171:80 backup;
}

server {
    listen 443 ssl;
    server_name 3.29.125.33;
    ssl_certificate /etc/ssl/certs/selfsigned.crt;
    ssl_certificate_key /etc/ssl/private/selfsigned.key;

    location / {
        root /usr/share/nginx/html;
        index index.html;
        # proxy_pass http://51.112.167.85:80;
        # proxy_pass http://backend_servers;

    }
}

server {
    listen 80;
```

```
task8_nginx_restart
[ec2-user@ip-10-0-10-51 ~]$ sudo vim /etc/nginx/nginx.conf
[ec2-user@ip-10-0-10-51 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-51 ~]$ |
```

task8_load_balancer_web1



task8_load_balancer_web2



Welcome to My Web Server

Hostname: myapp-webserver

Private IP: 10.0.10.9

Public IP: 3.29.244.171

Public DNS:

Task 9 — Configure high availability with backup servers

task9_nginx_conf_ha_web1_primary

```
ec2-user@ip-10-0-10-51:~$ pid /run/nginx.pid;
events {
    worker_connections 1024;
}

http {
    log_format main '$remote_addr - $remote_user [$time_local] "$request"'
                  '$status $body_bytes_sent "$http_referer"'
                  '"$http_user_agent" "$http_x_forwarded_for"';
    access_log /var/log/nginx/access.log main;

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

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

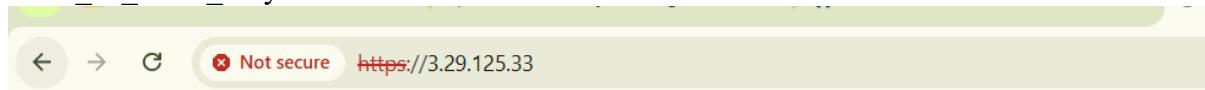
    upstream backend_servers {
        server 51.112.167.85:80;
        server 3.29.244.171:80 backup;
    }

    server {
        listen 443 ssl;
        server_name 3.29.125.33;
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
        ssl_certificate_key /etc/ssl/private/selfsigned.key;

        location / {
            proxy_pass http://backend_servers;
            proxy_set_header Host $host;
            proxy_set_header X-Real-IP $remote_addr;
        }
    }

    server {
-- INSERT --
```

task9_ha_web1_only



Welcome to My Web Server

Hostname: myapp-webserver

Private IP: 10.0.10.254

Public IP: 51.112.167.85

Public DNS:

task9_nginx_conf_ha_web2_primary

```
tcp_nopush      on;
keepalive_timeout 65;
types_hash_max_size 4096;

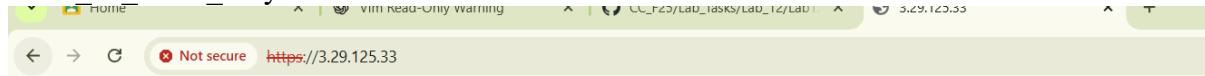
include          /etc/nginx/mime.types;
default_type     application/octet-stream;

upstream backend_servers {
    server 51.112.167.85:80 backup;
    server 3.29.244.171:80;
}

server {
    listen 443 ssl;
    server_name 3.29.125.33;
    ssl_certificate /etc/ssl/certs/selfsigned.crt;
    ssl_certificate_key /etc/ssl/private/selfsigned.key;

    location / {
        proxy_pass http://backend_servers;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
    }
}
```

task9 ha web2 only



Welcome to My Web Server

Hostname: myapp-webserver

Private IP: 10.0.10.9

Public IP: 3.29.244.171

Public DNS:

Task 10 — Enable Nginx caching

```
task10_nginx_conf_cache
default_type application/octet-stream;

upstream backend_servers {
    server 51.112.167.85:80;
    server 3.29.244.171:80;
}

server {
    listen 443 ssl;
    server_name 3.29.125.33;
    ssl_certificate /etc/ssl/certs/selfsigned.crt;
    ssl_certificate_key /etc/ssl/private/selfsigned.key;

    location / {
        #       root /usr/share/nginx/html;
        #       index index.html;
        #       proxy_pass http://<web-1-public-ip>:80;
        proxy_pass http://backend_servers;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_cache my_cache;
        proxy_cache_valid 200 60m;
        proxy_cache_key "$scheme$request_uri";
        add_header X-Cache-Status $upstream_cache_status;
    }
}

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

:wq!
```

task10_nginx_restart

```
[ec2-user@ip-10-0-10-51 ~]$ sudo vim /etc/nginx/nginx.conf
[ec2-user@ip-10-0-10-51 ~]$ sudo systemctl restart nginx
[ec2-user@ip-10-0-10-51 ~]$ |
```

task10_cache_miss

The screenshot shows the NetworkMiner tool interface. At the top, it says "Welcome to My Web Server". Below that, there are several labels: "Hostname: myapp-webserver", "Private IP: 10.0.10.9", "Public IP: 3.29.244.171", and "Public DNS:". The main part of the screen is a table showing network traffic. A row for a request to "?v=2" is selected. The Headers section shows:

Name	Headers	Payload	Preview	Response	Initiator	Timing
?v=2	Accept-Ranges: bytes Connection: keep-alive Content-Length: 156 Content-Type: text/html; charset=UTF-8 Date: Tue, 06 Jan 2026 14:31:11 GMT Etag: "9c-647b0d1dc606d" Last-Modified: Tue, 06 Jan 2026 04:43:40 GMT Server: nginx/1.28.0 X-Cache-Status: MISS			text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8		

At the bottom of the table, it says "3 requests | 1.3 kB transferred | Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8".

task10_cache_hit

The screenshot shows the NetworkMiner tool interface. At the top, it says "Welcome to My Web Server". Below that, there are several labels: "Hostname: myapp-webserver", "Private IP: 10.0.10.254", "Public IP: 51.112.167.85", and "Public DNS:". The main part of the screen is a table showing network traffic. A row for a request to "3.29.125.33" is selected. The Headers section shows:

Name	Headers	Preview	Response	Initiator	Timing
3.29.125.33	Accept-Ranges: bytes Connection: keep-alive Content-Length: 159 Content-Type: text/html; charset=UTF-8 Date: Tue, 06 Jan 2026 14:23:53 GMT Etag: "9f-647b01fda2ea0" Last-Modified: Tue, 06 Jan 2026 03:53:54 GMT Server: nginx/1.28.0 X-Cache-Status: HIT			3.29.125.33:443	500 ms 1,000 ms 1,500 ms 2,000 ms 2,500 ms 3,000 ms 3,500 ms 4,000 ms 4,500 ms 5,000 ms 5,500 ms 6,000 ms 6,500 ms 7,000 ms 7,500 ms

task10_cache_directory

```
[ec2-user@ip-10-0-10-51 ~]$ sudo ls -la /var/cache/nginx/
total 0
drwx----- 5 nginx root 33 Jan 6 14:31 .
drwxr-xr-x 9 root root 101 Jan 6 14:20 ..
drwx----- 3 nginx nginx 16 Jan 6 14:22 4
drwx----- 3 nginx nginx 16 Jan 6 14:31 8
drwx----- 3 nginx nginx 16 Jan 6 14:30 e
[ec2-user@ip-10-0-10-51 ~]$ |
```

Cleanup

cleanup_destroy_prompt

```
@23-22411-061-rgb ~ ~/Lab12 $ terraform destroy
data.http.my_ip: Reading...
data.http.my_ip: Read complete after 0s [id=https://icanhazip.com]
module.myapp-web-2.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-2]
module.myapp-web-1.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-1]
module.myapp-webserver.aws_key_pair.ssh-key: Refreshing state... [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Refreshing state... [id=vpc-0410b9b8db6f8d9db]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Refreshing state... [id=igw-07a753edb12402b67]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Refreshing state... [id=subnet-05532ac0c39950c21]
module.myapp-web-2.aws_security_group.web_sg: Refreshing state... [id=sg-03c9255a660e5dad8]
module.myapp-webserver.aws_security_group.web_sg: Refreshing state... [id=sg-0f97f72e6feb675cf]
module.myapp-web-1.aws_security_group.web_sg: Refreshing state... [id=sg-0f56c1b91ea052f64]
module.myapp-subnet.aws_default_route_table.main_rt: Refreshing state... [id=rtb-07ecf971dd3af53f6]
module.myapp-web-2.aws_instance.myapp-server: Refreshing state... [id=i-0a58ca8b54da9c9cb]
module.myapp-web-1.aws_instance.myapp-server: Refreshing state... [id=i-0b44a5df236217f77]
module.myapp-webserver.aws_instance.myapp-server: Refreshing state... [id=i-031924a29cf02bc97]
```

cleanup_destroy_complete

```
module.myapp-web-1.aws_instance.myapp-server: Destruction complete after
module.myapp-web-1.aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-2]
module.myapp-web-1.aws_security_group.web_sg: Destroying... [id=sg-0f56c1b91ea052f64]
module.myapp-web-1.aws_key_pair.ssh-key: Destruction complete after 1s
module.myapp-web-1.aws_security_group.web_sg: Destruction complete after 1s
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0a58ca8b54da9c9cb]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-0b44a5df236217f77]
module.myapp-webserver.aws_instance.myapp-server: Still destroying... [id=i-031924a29cf02bc97]
module.myapp-webserver.aws_instance.myapp-server: Destruction complete after
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id=subnet-05532ac0c39950c21]
module.myapp-webserver.aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-1]
module.myapp-webserver.aws_security_group.web_sg: Destroying... [id=sg-03c9255a660e5dad8]
module.myapp-web-1.aws_key_pair.ssh-key: Destruction complete after 1s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 1s
module.myapp-webserver.aws_security_group.web_sg: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-0410b9b8db6f8d9db]
aws_vpc.myapp_vpc: Destruction complete after 0s
```

Destroy complete! Resources: 13 destroyed.

```
@23-22411-061-rgb ~ ~/Lab12 $ |
```



cleanup_state_empty

```
Destroy complete! Resources: 13 destroyed.
@23-22411-061-rgb ~ ~/Lab12 $ cat terraform.tfstate
```

```
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 125,
  "lineage": "a6ca3fb0-cb54-04e9-f61d-30986271736f",
  "outputs": {},
  "resources": [],
  "check_results": null
}
```

```
@23-22411-061-rgb ~ ~/Lab12 $ |
```

cleanup_final_files

```
total 61816
drwxrwxr-x 5 codespace codespace 4096 Jan  6 14:38 .
drwxr-x--- 1 codespace codespace 4096 Jan  6 13:39 ..
drwxr-xr-x 4 codespace codespace 4096 Jan  5 13:44 .terraform
-rw-r--r-- 1 codespace codespace 2422 Jan  5 09:21 .terraform.lock.hcl
-rw-rw-r-- 1 codespace codespace 945 Jan  6 03:46 apache.sh
drwxr-xr-x 3 codespace codespace 4096 Jan  2 23:18 aws
-rw-rw-r-- 1 codespace codespace 63189473 Jan  5 09:24 awscli2.zip
-rw-rw-r-- 1 codespace codespace 2341 Jan  5 18:10 entry-script.sh
-rw-rw-r-- 1 codespace codespace 123 Jan  5 09:11 locals.tf
-rw-rw-r-- 1 codespace codespace 1579 Jan  6 04:27 main.tf
drwxrwxr-x 4 codespace codespace 4096 Jan  5 13:48 modules
-rw-rw-r-- 1 codespace codespace 262 Jan  6 08:47 outputs.tf
-rw-rw-r-- 1 codespace codespace 183 Jan  6 14:38 terraform.tfstate
-rw-rw-r-- 1 codespace codespace 42360 Jan  6 14:37 terraform.tfstate.backup
-rw-rw-r-- 1 codespace codespace 219 Jan  5 09:13 terraform.tfvars
-rw-rw-r-- 1 codespace codespace 198 Jan  5 09:08 variables.tf
```
