



CLOUD COMPUTING LAB **BSE (V-B)**

Submitted By:

Musfira Farooq

Roll No:

2023-BSE-045

Submitted To:

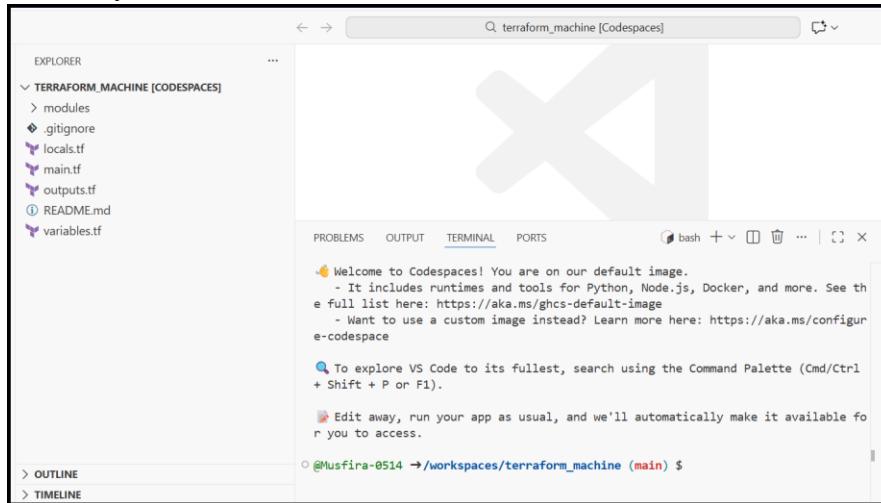
Sir Muhammad Shoaib

LAB 14(openended lab)

Terraform + Ansible: Dynamic Inventory, Roles & Automated Nginx/PHP & Docker Deployment

Task 0 – Lab Setup (Codespace & GH CLI)

Codespace successfully launched



```
Welcome to Codespaces! You are on our default image.
- It includes runtimes and tools for Python, Node.js, Docker, and more. See the full list here: https://aka.ms/gchcs-default-image
- Want to use a custom image instead? Learn more here: https://aka.ms/configure-codespace

To explore VS Code to its fullest, search using the Command Palette (Cmd/Ctrl + Shift + P or F1).

Edit away, run your app as usual, and we'll automatically make it available for you to access.

@Musfira-0514 →/workspaces/terraform_machine (main) $
```

Environment verification inside Codespace

```
@Musfira-0514 →/workspaces/terraform_machine (main) $ aws --version
aws-cli/2.32.30 Python/3.13.11 Linux/6.8.0-1030-azure exe/x86_64.ubuntu.24
Terraform v1.14.3
on linux_amd64
ansible [core 2.16.3]
  config file = None
  configured module search path = ['~/.home/codespace/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/codespace/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Nov 6 2025, 13:44:16) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.6
  libyaml = True
  github.com
  ✓ Logged in to github.com account Musfira-0514 (GITHUB_TOKEN)
  - Active account: true
  - Git operations protocol: https
  - Token: ghu_*****
```

AWS CLI configured inside Codespace

```
DEPRECATION: The default value of the 'cli_config_file' parameter has changed from '~/.aws/config' to '~/.aws/config' in version 2.0.0. To maintain compatibility with the previous behavior, set the 'cli_config_file' parameter to '~/.aws/config' when using the 'aws' command. This parameter will be removed in a future version of the AWS CLI.
@Musfira-0514 →/workspaces/terraform_machine (main) $ aws sts get-caller-identity
{
  "UserId": "AIDAVSVUK5OYDSMI7QJNW",
  "Account": "383704034224",
  "Arn": "arn:aws:iam::383704034224:user/ansible-user"
}
@Musfira-0514 →/workspaces/terraform_machine (main) $
```

Task 1 – SSH Key Generation & Initial Terraform Apply

SSH key status before generation

```
@Musfira-0514 →/workspaces/terraform_machine (main) $ ls ~/.ssh
@Musfira-0514 →/workspaces/terraform_machine (main) $
```

SSH key generation command execution

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ 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:
The key's randomart image is:
++-[ED25519 256]--+
|B.. B@o. . . .
|.+ = . . o o
|... o o .+
|... . o . .
|. . . S .
| . o + + . .
|= + o.o .
| + o++...
|...*+=*o
+---[SHA256]---+
○ @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

SSH key successfully created

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ ls -la ~/.ssh
total 20
drwx----- 2 codespace codespace 4096 Jan  6 21:00 .
drwxr-x--- 1 codespace codespace 4096 Jan  6 20:59 ..
-rw----- 1 codespace codespace  419 Jan  6 21:00 id_ed25519
-rw-r--r-- 1 codespace codespace 109 Jan  6 21:00 id_ed25519.pub
○ @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform variables file created

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ touch terraform.tfvars
● @Musfira-0514 → /workspaces/terraform_machine (main) $ cd /workspaces/terraform_machine
touch terraform.tfvars
ls -la terraform.tfvars
-rw-rw-rw- 1 codespace codespace 0 Jan  6 21:04 terraform.tfvars
○ @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform variables configuration

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ nano terraform.tfvars
● @Musfira-0514 → /workspaces/terraform_machine (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 = "dev"
instance_type = "t3.micro"
public_key = "~/.ssh/id_ed25519.pub"
private_key = "~/.ssh/id_ed25519"
○ @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform initialization

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ terraform init
Initializing the backend...
Initializing modules...
- myapp-subnet in modules/subnet
- myapp-webserver in modules/webserver
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Finding latest version of hashicorp/http...
- Installing hashicorp/aws v6.27.0...
- Installed hashicorp/aws v6.27.0 (signed by HashiCorp)
- Installing hashicorp/http v3.5.0...
- Installed hashicorp/http v3.5.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

```

Terraform apply creating two EC2 instances

```

module.myapp-webserver[1].aws_security_group.web_sg: Creating...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creation complete after 0s [id=igw-0239a2c2075c24d1d]
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-0eab18bea49058ff4]
module.myapp-webserver[1].aws_security_group.web_sg: Creation complete after 3s [id=sg-01af2ad11520e44a7]
module.myapp-webserver[0].aws_security_group.web_sg: Creation complete after 3s [id=sg-0a92e979754c7e311]
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-0835f25fdcef512a8]
module.myapp-webserver[0].aws_instance.myapp-server: Creating...
module.myapp-webserver[1].aws_instance.myapp-server: Creating...
module.myapp-webserver[0].aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Creation complete after 12s [id=i-08129fb61fc344b81]
module.myapp-webserver[1].aws_instance.myapp-server: Creation complete after 12s [id=i-090ef3643a108019b]

Apply complete! Resources: 10 added, 0 changed, 0 destroyed.

Outputs:

webserver_public_ips = [
  "158.252.94.86",
  "3.28.163.208",
]
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform outputs showing public IPs

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "158.252.94.86",
  "3.28.163.208",
]

```

Task 2 – Static Ansible Inventory Configuration

Ansible installation verification

```

These apps are now globally available
- ansible
- ansible-config
- ansible-console
- ansible-doc
- ansible-galaxy
- ansible-inventory
- ansible-playbook
- ansible-pull
- ansible-test
- ansible-vault
done! ✨
ansible [core 2.20.1]
  config file = None
  configured module search path = ['/home/codespace/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/local/py-utils/venvs/ansible-core/lib/python3.12/site-packages/ansible
  ansible collection location = /home/codespace/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/local/py-utils/bin/ansible
  python version = 3.12.1 (main, Nov 27 2025, 10:47:52) [GCC 13.3.0] (/usr/local/py-utils/venvs/ansible-core/bin/python)
  jinja version = 3.1.6
  pyyaml version = 6.0.3 (with libyaml v0.2.5)
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform output used for inventory

```

pyyaml version = 6.0.3 (with libyaml v0.2.5)
@Musfira-0514 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "158.252.94.86",
  "3.28.163.208",
]
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Initial Ansible hosts file created

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ touch hosts
ls -la hosts
-rw-rw-rw- 1 codespace codespace 0 Jan  6 21:12 hosts
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Hosts file before adding SSH arguments

```
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ nano hosts
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ cat hosts

158.252.94.86 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519
3.28.163.208 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519
○ @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ █
(0) o
```

Ansible ping failure due to missing SSH args

```
@Musfira-0514 ➔ /workspaces/terraform_machine (main) $ @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[ERROR]: Task failed: Failed to connect to the host via ssh: Host key verification failed.
Origin: <adhoc 'ping' task>

{'action': 'ping', 'args': {}, 'timeout': 0, 'async_val': 0, 'poll': 15}

3.28.163.208 | UNREACHABLE! => {
    "changed": false,
    "msg": "Task failed: Failed to connect to the host via ssh: Host key verification failed.",
    "unreachable": true
}
158.252.94.86 | UNREACHABLE! => {
    "changed": false,
    "msg": "Task failed: Failed to connect to the host via ssh: Host key verification failed.",
    "unreachable": true
}
```

Hosts file updated with common SSH arguments

```
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ nano hosts
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ cat hosts
158.252.94.86 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
3.28.163.208 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
```

Successful Ansible ping to EC2 instances

```
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '3.28.163.208' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.163.208 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
158.252.94.86 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
```

Task 3 – Scaling Infrastructure to Three Instances

Terraform updated to launch three instances

```
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

  # Loop count
  count          = 3
  # Use count.index to differentiate instances
  instance_suffix = count.index
}
○ @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ █
(0) o
```

Terraform apply showing three instances created

```
@Musfira-0514 → /workspaces/terraform_machine (main) $ terraform apply -auto-approve

Changes to Outputs:
  ~ webserver_public_ips = [
      # (1 unchanged element hidden)
      "3.28.163.208",
      + (known after apply),
    ]
  module.myapp-webserver[2].aws_key_pair.ssh-key: Creating...
  module.myapp-webserver[2].aws_security_group.web_sg: Creating...
  module.myapp-webserver[2].aws_key_pair.ssh-key: Creation complete after 1s [id=dev-serverkey-2]
  module.myapp-webserver[2].aws_security_group.web_sg: Creation complete after 3s [id=sg-03aae5bcdf954fc8]
  module.myapp-webserver[2].aws_instance.myapp-server: Creating...
  module.myapp-webserver[2].aws_instance.myapp-server: Still creating... [00m10s elapsed]
  module.myapp-webserver[2].aws_instance.myapp-server: Creation complete after 13s [id=i-0acd4b3bba0537f7]

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

Outputs:

webserver_public_ips = [
  "158.252.94.86",
  "3.28.163.208",
  "51.112.167.59",
]

@Musfira-0514 → /workspaces/terraform_machine (main) $
```

Terraform output displaying three IPs

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ terraform output
  webserver_public_ips = [
    "158.252.94.86",
    "3.28.163.208",
    "51.112.167.59",
  ]
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Inventory grouped into EC2 and droplet hosts

```
GNU nano 7.2
[ec2]
158.252.94.86
3.28.163.208

[ec2:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_ssh_common_args=' -o StrictHostKeyChecking=no'

[droplet]
51.112.167.59

[droplet:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
```

Ansible ping to EC2 group

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ nano hosts
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible ec2 -i hosts -m ping
[WARNING]: Host '3.28.163.208' is using the discovered Python interpreter at '/usr/bin/python3.9', interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/interpreter_discovery.html for more information.
3.28.163.208 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9', interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/interpreter_discovery.html for more information.
158.252.94.86 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Ansible ping to single EC2 IP

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible 158.252.94.86 -i hosts -m ping
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9'
n interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/a
rpreter_discovery.html for more information.
158.252.94.86 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Ansible ping to droplet group

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible droplet -i hosts -m ping
[WARNING]: Host '51.112.167.59' is using the discovered Python interpreter at '/usr/bin/python3.9'
n interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/a
rpreter_discovery.html for more information.
51.112.167.59 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Ansible ping to all hosts

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '51.112.167.59' is using the discovered Python interpreter at '/usr/bin/python3.9'
erpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/int
51.112.167.59 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.163.208' is using the discovered Python interpreter at '/usr/bin/python3.9', r
preter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/int
3.28.163.208 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9', r
preter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/int
158.252.94.86 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Task 4 – Global Ansible Configuration & Nginx Deployment

Global ansible.cfg configuration

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
[default]
host_key_checking = False
interpreter_python = /usr/bin/python3
~
```

Hosts file without common SSH arguments

```
GNU nano 7.2
[ec2]
158.252.94.86
3.28.163.208

[ec2:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/ssh/id_ed25519

[droplet]
51.112.167.59

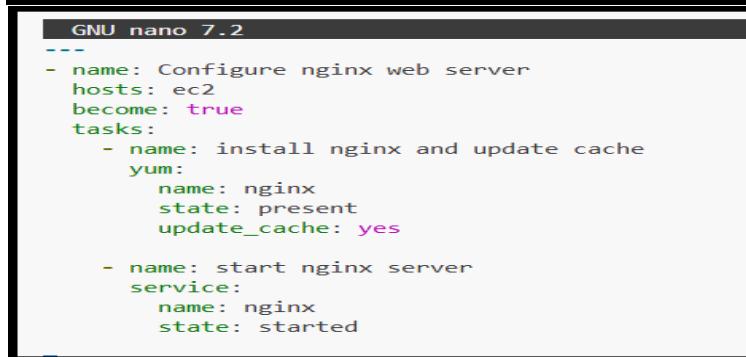
[droplet:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/ssh/id_ed25519
```

Successful Ansible ping using global config

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ vim ~/.ansible.cfg
● @Musfira-0514 → /workspaces/terraform_machine (main) $ nano hosts
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '51.112.167.59' is using the discovered Python interpreter at '/usr/bin/python3.9'
interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference\_appendices/int
51.112.167.59 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Host '3.28.163.208' is using the discovered Python interpreter at '/usr/bin/python3.9'
interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference\_appendices/int
3.28.163.208 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9'
interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference\_appendices/int
158.252.94.86 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.9"
  },
  "changed": false,
  "ping": "pong"
}
● @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Ansible playbook creation

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ touch my-playbook.yaml
ls -la my-playbook.yaml
-rw-rw-rw- 1 codespace codespace 0 Jan  6 21:47 my-playbook.yaml
● @Musfira-0514 → /workspaces/terraform_machine (main) $
```



```
GNU nano 7.2
---
- name: Configure nginx web server
  hosts: ec2
  become: true
  tasks:
    - name: install nginx and update cache
      yum:
        name: nginx
        state: present
        update_cache: yes

    - name: start nginx server
      service:
        name: nginx
        state: started
```

Nginx deployment playbook for EC2

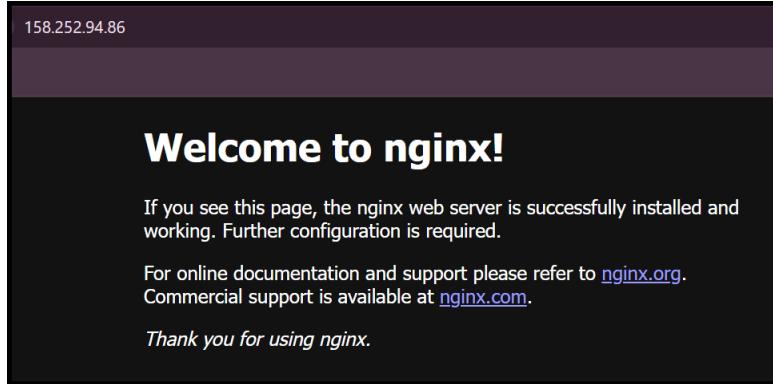
```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml
PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.163.208' is using the discovered Python interpreter at '/usr/bin/python3.9', but future install
interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference\_appendices/interpreter\_discovery.html
ok: [3.28.163.208]
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9', but future install
interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference\_appendices/interpreter\_discovery.html
ok: [158.252.94.86]

TASK [install nginx and update cache] ****
changed: [3.28.163.208]
changed: [158.252.94.86]

TASK [start nginx server] ****
changed: [3.28.163.208]
changed: [158.252.94.86]

PLAY RECAP ****
158.252.94.86 : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
3.28.163.208 : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Successful playbook execution on EC2



Nginx accessed via browser on EC2

```
GNU nano 7.2
---
- name: Configure nginx web server
  hosts: droplet
  become: true
  tasks:
    - name: install nginx and update cache
      yum:
        name: nginx
        state: present
        update_cache: yes

    - name: start nginx server
      service:
        name: nginx
        state: started
```

Nginx deployment playbook for droplet

```
● @Musfira-0514 →/workspaces/terraform_machine (main) $ nano my-playbook.yaml
● @Musfira-0514 →/workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '51.112.167.59' is using the discovered Python interpreter at '/usr/bin/python3.9', but future interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.28/reference_appendices/interpreter_discovery.html#interpreter-discovery
ok: [51.112.167.59]

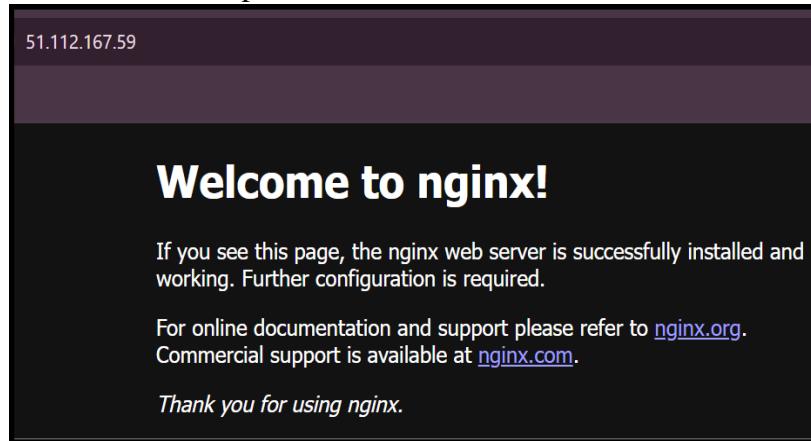
TASK [install nginx and update cache] ****
changed: [51.112.167.59]

TASK [start nginx server] ****
changed: [51.112.167.59]

PLAY RECAP ****
51.112.167.59 : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

● @Musfira-0514 →/workspaces/terraform_machine (main) $
```

Nginx accessed via browser on droplet



Task 5 – Single Target Group & HTTPS Preparation

Project-level ansible.cfg created

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ touch ansible.cfg
ls -la ansible.cfg
-rw-rw-rw- 1 codespace codespace 0 Jan  6 22:00 ansible.cfg
● @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Project ansible.cfg configuration

```
GNU nano 7.2
[defaults]
host_key_checking=False
interpreter_python = /usr/bin/python3
```

Terraform updated to single instance

```
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

  # Loop count
  count = 1
  # Use count.index to differentiate instances
  instance_suffix = count.index
}
```

Terraform apply for single EC2 instance

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ terraform apply
[...]
module.myapp-webserver[2].aws_instance.myapp-server: Destroying... [id=0-acd4b3bba0537f7c]
module.myapp-webserver[1].aws_instance.myapp-server: Destroying... [id=0-900ef3643a108019b]
module.myapp-webserver[2].aws_instance.myapp-server: Still destroying... [id=0-acd4b3bba0537f7c, 0m10s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m10s elapsed]
module.myapp-webserver[2].aws_instance.myapp-server: Still destroying... [id=0-acd4b3bba0537f7c, 0m20s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m20s elapsed]
module.myapp-webserver[2].aws_instance.myapp-server: Destruction complete after 30s
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m30s elapsed]
module.myapp-webserver[2].aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-1]
module.myapp-webserver[2].aws_security_group.web_sg: Destroying... [id=sg-03aae5bcd954fc86]
module.myapp-webserver[2].aws_key_pair.ssh-key: Destruction complete after 0s
module.myapp-webserver[2].aws_security_group.web_sg: Destruction complete after 1s
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m40s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m50s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m00s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Still destroying... [id=0-900ef3643a108019b, 0m10s elapsed]
module.myapp-webserver[1].aws_instance.myapp-server: Destruction complete after 1m0s
module.myapp-webserver[1].aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-1]
module.myapp-webserver[1].aws_security_group.web_sg: Destroying... [id=sg-01af2ad11520e44a7]
module.myapp-webserver[1].aws_key_pair.ssh-key: Destruction complete after 1s
module.myapp-webserver[1].aws_security_group.web_sg: Destruction complete after 1s

Apply complete! Resources: 0 added, 0 changed, 6 destroyed.

Outputs:
webserver_public_ips = [
  "158.252.94.86",
]
● @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Terraform output showing single IP

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "158.252.94.86",
]
● @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Inventory created for nginx group

```
GNU nano 7.2
[nginx]
158.252.94.86

[nginx:vars]
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_user=ec2-user
```

Playbook targeting nginx group

```
GNU nano 7.2
---
- name: Configure nginx web server
  hosts: nginx
  become: true
  tasks:
    - name: install nginx and update cache
      yum:
        name: nginx
        state: present
        update_cache: yes

    - name: install openssl
      yum:
        name: openssl
        state: present

    - name: start nginx server
      service:
        name: nginx
        state: started
        enabled: true
```

Successful playbook execution

```
● @lusfira-0514 → /workspaces/terraform_machine (main) $ nano hosts
● @lusfira-0514 → /workspaces/terraform_machine (main) $ nano my-playbook.yaml
● @lusfira-0514 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml
[WARNING]: Ansible is being run in a world writable directory (/workspaces/terraform_machine), ignoring it as an ansible/dev1/reference_appendices/config.html#cfg-in-world-writable-dir

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '158.252.94.86' is using the discovered Python interpreter at '/usr/bin/python3.9', but future interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html#cfg-in-world-writable-dir
ok: [158.252.94.86]

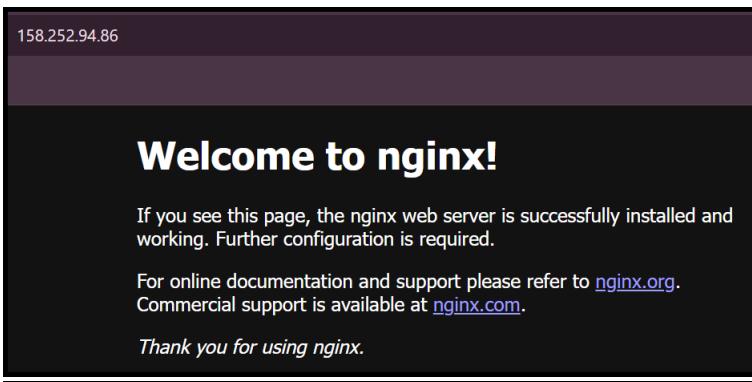
TASK [install nginx and update cache] ****
ok: [158.252.94.86]

TASK [install openssl] ****
ok: [158.252.94.86]

TASK [start nginx server] ****
changed: [158.252.94.86]

PLAY RECAP ****
158.252.94.86 : ok=4    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Nginx verified in browser



The screenshot shows a browser window with the IP address 158.252.94.86 in the address bar. The page content is the standard Nginx welcome message: "Welcome to nginx!". Below the title, it says, "If you see this page, the nginx web server is successfully installed and working. Further configuration is required." It also provides links to nginx.org and nginx.com. At the bottom, it says, "Thank you for using nginx."

Task 6 – SSL Certificate Automation

SSL configuration added to Ansible playbook

```

- name: Configure SSL certificates
  hosts: nginx
  become: true
  tasks:
    - name: Create SSL private directory
      file:
        path: /etc/ssl/private
        state: directory
        mode: '0700'
    - name: Create SSL certs directory
      file:
        path: /etc/ssl/certs
        state: directory
        mode: '0755'
    - name: Get IMDSv2 token
      uri:
        url: "http://169.254.169.254/latest/api/token"
        method: PUT
        headers:
          X-aws-ec2-metadata-token-ttl-seconds: "3600"
        return_content: yes
      register: imds_v2_token
    - name: Get current public IP
      uri:
        url: "http://169.254.169.254/latest/meta-data/public-ipv4"
        headers:
          X-aws-ec2-metadata-token: "{{ imds_v2_token.content }}"
        return_content: yes
      register: public_ip
    - name: Show current public IP
      debug:
        msg: "Public IP: {{ public_ip.content }}"
    - name: Generate self-signed SSL certificate
      command: >
        openssl req -x509 -nodes -days 365
        -newkey rsa:2048
        -keyout /etc/ssl/private/selfsigned.key
        -out /etc/ssl/certs/selfsigned.crt
        -subj "/O={{ public_ip.content }}"
        -addext "subjectAltName=IP:{{ public_ip.content }}"
        -addext "basicConstraints=CA:FALSE"
        -addext "keyUsage=digitalSignature, keyEncipherment"
        -addext "extendedKeyUsage=serverAuth"
      args:
        creates: /etc/ssl/certs/selfsigned.crt
  main@Musfira-0514 ~>/workspaces/terraform_machine (main) $ 
  main@Musfira-0514 ~>/workspaces/terraform_machine (main) $ 

```

SSL playbook executed successfully

```

@Musfira-0514 ~>/workspaces/terraform_machine (main) $ ssh ec2-user@158.252.94.86 -i ~/.ssh/id_ed25519
  ~~~ V~' `-'>
  ~~~~ /`_
  ~~~._. /`_
  _/ _/`_
  /m/'`_
Last login: Tue Jan  6 22:19:01 2026 from 4.240.18.228
[ec2-user@ip-10-0-10-211 ~]$ sudo cat /etc/ssl/certs/selfsigned.crt
-----BEGIN CERTIFICATE-----
MIIDPzCCAiegAwIBAgIUXzbn/h03QWognTz+w4r1pbLDJxkwDQYJKoZIhvcNAQEL
BQAwGDEWMBQGA1UEAwNMTU4LjI1Mi45NC44NjAeFw0yNjAxMDYyMjE5MDJaFw0y
NzAxMDYyMjE5MDJaMBgxFjAUBgNVBAMMDTE10C4yNTIuOTQuODYwggEiMA0GCSqG
SIb3DQEBAQAA4IBDwAwggEKAoIBAQDC1n1dU0VU7Vwe5mmQ8ct2SkgU7uKuUIe
V1Mz0v+/On10ARLbGh/kZkhyHdiq9MUZo+Tn98g9629Gj2Q3d70fY1lL0ezB41
cCHE/1jtTwaGhdGn+w4ycgzN7wQ91eASGybgeJQCclsPFAKZq9LPYQ/JiX0cB
OokF++/MDjtZUt6+R7vJBn6y/q1DrhmHaef50DG1aCxx0KghbRLL45dnba0Z6jJ
KUDXHd5U04XKXGfsUF1/AJ/F0E8I7TRSpTVAQCCjZ6j9bYrhMFnbHu0ny8GAj6r
WdfyNw2u9cva79KTK9/anW/ellwYBCTsxai8Gh2gz5cdoZ5+0FPAgMBAAGjgYAw
fjAddBgnVHQ4EfqQUbhYFn98Tn8M0PJfrQ8/3jpyP0ScwflwYDVR0jBBywFoAUBHF
n98Tn8M0PJfrQ8/3jpyP0ScwDwYDVR0RBAgwBocEnvxeVjAJBgnVHRMEAjAAMAsG
A1UdDwQEAWIFoDABgNVHUEDDAKBgrBgfFBQcDATANBgkqhkiG9w0BAQsFAAO
AQEAGne4rcwaPHTdJmmrz5eJTkvUvSriZ+rfauiuZhXgwLJmLr7Ya7ls2frkhImr
TGVMz0P5qs5UBf8/ba5e8Q00giFs5LkIEFwh7kJMKZYxxBArm3AVHR7abuLwY03
GxsA/OuGD9g61SeoNoRiRo0IueXy8K0rBvw+I8G7V+xCMmgwlaMvquZ9f6ZydH+
i0miKKOnaPNcOdNMxP76eSV2hF7/7i/8Mhrd5cWghN3kyC6MsDANI0XXG2jrSbeV
zDSAuuuEEyOFBoKqPmlc9Zc4i9va93WNGZ4PDMtMYI+3DmIghdNoVk/1K+sksp+q
CJWl6rGrzqX5M+sytk+t+3d9Sg==
-----END CERTIFICATE-----
[ec2-user@ip-10-0-10-211 ~]$ 

```

```
[ec2-user@ip-10-0-10-211 ~]$ sudo cat /etc/ssl/private/selfsigned.key
-----BEGIN PRIVATE KEY-----
MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQDC1n1dU0VU7Vwe
5mmQ8ct2SkgU7uKuUteV1Mz0v+/On10ARLaGh/kZKhyHdiq9WUzo+Tn98G96z9G
j2Q3d70Fy1LilLozb41ccHE/1jtTwaghdgn6+w4cggN7ww9q1eASGvbgeJQC1c
sPFAKZq9lPYQ/JixOcBookF++/MJqtZUt6+R7vJBn6y/qIDrhmAefS0DG1aCxk
OkGhBRLL45dIbaWz6jJKUDXHd5U4KXGFSUf1/AJ/F08EI7TRSpVTAVQccjZ6j9
bYrhMFnbHui0ny8GAj6rWDFyNw2u9cva79KTk9/anW/e1LwYBCTSzaiP8GH2gz5c
doZ5+0FPAgMBAECggEAHJ//n1d4TDBtOtr2Tw7BKjLAEJBMuuZZ6PRCdjg428L/
oHxx65ov+1Hurd1ltkREXL6ACHaVf84ap0a@3hNY8Yt8019+6Kmp4K10Xv1RX1gg
aM4r+TcTzGsrIH4I+Q0CNFzrTmVIZC1jimUyPj//MTKPY6Qj7cChbcRIU1k@01nP
Pc+Nfry//SE6p2w9GY6Ve6SOH3g9DduX12cN5uP014UggSJ4LN9q4PQLL/rejU0S
QRkvBBL9X77jZA7TOC4PebgRnsouggkm7ptQpCPGwfPsr1j099sxRoJikd7WNFv
EzzFogHGTMaJBRNeNSFrH/pju10T7TMZAAuFnmaGVQKBgQDw54W49Lz1kVTsp2be
JPbQJ8nLmz3/NEG8kmDo+AkkMze5vTuc0Izvc4mld5jKT4QkP16gkDMZAE0Im9
SmL61y19CnS9gbhQFCZ+B1Jkkf0CtvLT6gNpx1hGCV2eELAGVXT/aqQOR3tWoH
QDyrevDS05TTTVizW15puuhjEwkBqQDPC/04jPCCNRH17BNv9utRw7U0Yb8b0ThU
QL4Z9Ae7Ab219nQ8YvklwJml1joNA1MALGc97gcYc20mv3Tikp4uIRp540HcRmkPV
gf2vNj206hPT0pDmU08W74gL4j0kYEwc30AoaQ/QTHegf65dGadFK7q68RFdRft
FHkjjiHOUVQKBgF18dqWgxir8xZGfxOWQrr1i/tJi0hxgVmBtB6//mc1Hjm15Jfu
QVPUcwhbNsGvhVFKLat6rQduFl+afUqk7gZ7Dq1l0wnL3gGq00yqeSPnCCy3oT
1H5j66BdgcxfQATlabLY1TeepyvGO/B73w1QwzCkhdfu/veIOiLaNEJAoGAPeYe
sSSprLmu5Fy2JW4Tkglx3VQvt1hV9/3NTR6KwvjBdxsbAB0d8GCA0g1lysWUZ
A5qbaPriP8KKcadpPbrdHwJ6h/evugAf3+mfnhtdyTMku8T0M0xWPfcYFnGnxt1w
DKVgmaWSw11fxdMuMT70158EivcmFnbmAtuUPUCgYAdkPiCp7d1u53dqmKwbj1o
K2Wgmy07D3ltmA8wC12nRfQ8CNDw+0wNpwe9eZkuu1FFDYgjI85dgHrwss1h9zbT
mWS/C5ZxKZ0ZadnveZE7qvNAuX8Dows/PLMeDgTENhXwwPaWDbzGT+DcbR9PfxH
+nT51AqqrMyTpRtd08UPig==

-----END PRIVATE KEY-----
[ec2-user@ip-10-0-10-211 ~]$
```

Task 7 – PHP Frontend Deployment

Files and templates directory created

```
./files: ✓
index.php ✓

./modules:
subnet  webserver

./modules/subnet:
main.tf  outputs.tf  variables.tf

./modules/webserver:
main.tf  outputs.tf  variables.tf

./templates:
nginx.conf.j2 ✓
@Musfira-0514 → /workspaces/terraform_machine (main) $
```

PHP index file content

```
GNU nano 7.2
<?php
// Get hostname
$hostname = gethostname();

// Deployment date
$deployed_date = date("Y-m-d H:i:s");

// Metadata base URL
$metadata_base = "http://169.254.169.254/latest/";

// Function to get IMDSv2 token
function getImdsV2Token() {
    $ch = curl_init("http://169.254.169.254/latest/api/token");
    curl_setopt_array($ch, [
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_CUSTOMREQUEST => "PUT",
        CURLOPT_HTTPHEADER => [
            "X-aws-ec2-metadata-token-ttl-seconds: 21600"
        ],
        CURLOPT_TIMEOUT => 2
    ]);

    $token = curl_exec($ch);
    curl_close($ch);

    return $token ?: null;
}

// Function to fetch metadata using token
function getMetadata($path, $token) {
    $ch = curl_init($path);
    curl_setopt_array($ch, [
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_HTTPHEADER => [
            "X-aws-ec2-metadata-token: $token"
        ]
    ]);

    $response = curl_exec($ch);
    curl_close($ch);

    return $response ?: null;
}
```

^G Help ^O Write Out ^W Where Is ^K Cut
^X Exit ^R Read File ^V Replace ^U Paste ^T Exec
^J Just

```

GNU nano 7.2
function getMetadata($path, $token) {
    $url = "http://169.254.169.254/latest/meta-data/" . $path;

    $ch = curl_init($url);
    curl_setopt_array($ch, [
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_HTTPHEADER => [
            "X-aws-ec2-metadata-token: $token"
        ],
        CURLOPT_TIMEOUT => 2
    ]);

    $value = curl_exec($ch);
    curl_close($ch);

    return $value ?: "N/A";
}

// Fetch token
$token = getImdsV2Token();

// Fetch metadata only if token is available
$instance_id = $token ? getMetadata("instance-id", $token) : "N/A";
$private_ip = $token ? getMetadata("local-ipv4", $token) : "N/A";
$public_ip = $token ? getMetadata("public-ipv4", $token) : "N/A";
$public_dns = $token ? getMetadata("public-hostname", $token) : "N/A";
?>
<!DOCTYPE html>
<html>
<head>

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

GNU nano 7.2                                         files/index.php *
.info a {
    color: white;          /* same as other values */
    text-decoration: none; /* remove underline */
    font-weight: normal;
}

.info a:hover {
    text-decoration: underline; /* optional: underline on hover */
}
</style>
</head>
<body>
<div class="container">
    <h1> Nginx Front End Web Server </h1>

    <div class="info"><span class="label">Hostname:</span> <?= htmlspecialchars($hostname) ?></div>
    <div class="info"><span class="label">Instance ID:</span> <?= htmlspecialchars($instance_id) ?></div>
    <div class="info"><span class="label">Private IP:</span> <?= htmlspecialchars($private_ip) ?></div>
    <div class="info"><span class="label">Public IP:</span> <?= htmlspecialchars($public_ip) ?></div>
    <div class="info"><span class="label">Public DNS:</span>
        <a href="https://<?= htmlspecialchars($public_dns) ?>" target="_blank">
            https://<?= htmlspecialchars($public_dns) ?></a>
    </div>
    <div class="info"><span class="label">Deployed:</span> <?= $deployed_date ?></div>
    <div class="info"><span class="label">Status:</span>  Active and Running</div>
    <div class="info"><span class="label">Managed By:</span> Terraform + Ansible</div>
</div>
</body>
</html>

```

Nginx virtual host template for PHP

```

GNU nano 7.2
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log notice;
pid /run/nginx.pid;

events {
    worker_connections 1024;
}

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

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

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

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

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

GNU nano 7.2
server {
    listen 443 ssl;
    server_name {{ server_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.php index.html index.htm;
        # proxy_pass http://158.252.94.241:80;
        # proxy_pass http://backend_servers;

        # ● This block is necessary for Php Website
        location ~ \.php$ {
            include fastcgi_params;
            fastcgi_pass unix:/run/php-fpm/www.sock;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        }
    }
}

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

```

Web deployment playbook configuration

```

GNU nano 7.2
- name: Deploy Nginx website and configuration files
  hosts: nginx
  become: true
  tasks:
    - name: install php-fpm and php-curl
      yum:
        name:
          - php-fpm
          - php-curl
        state: present

    - name: Copy website files
      copy:
        src: files/index.php
        dest: /usr/share/nginx/html/index.php
        owner: nginx
        group: nginx
        mode: '0644'

    - name: Copy nginx.conf template
      template:
        src: templates/nginx.conf.j2
        dest: /etc/nginx/nginx.conf
        owner: root
        group: root
        mode: '0644'

    - name: Restart nginx
      service:

```

Successful PHP deployment via Ansible

```
@Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml

TASK [Show current public IP] ****
ok: [158.252.94.86] => {
    "msg": "Public IP: 158.252.94.86"
}

TASK [Generate self-signed SSL certificate] ****
ok: [158.252.94.86]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [158.252.94.86]

TASK [install php-fpm and php-curl] ****
ok: [158.252.94.86]

TASK [Copy website files] ****
ok: [158.252.94.86]

TASK [Copy nginx.conf template] ****
changed: [158.252.94.86]

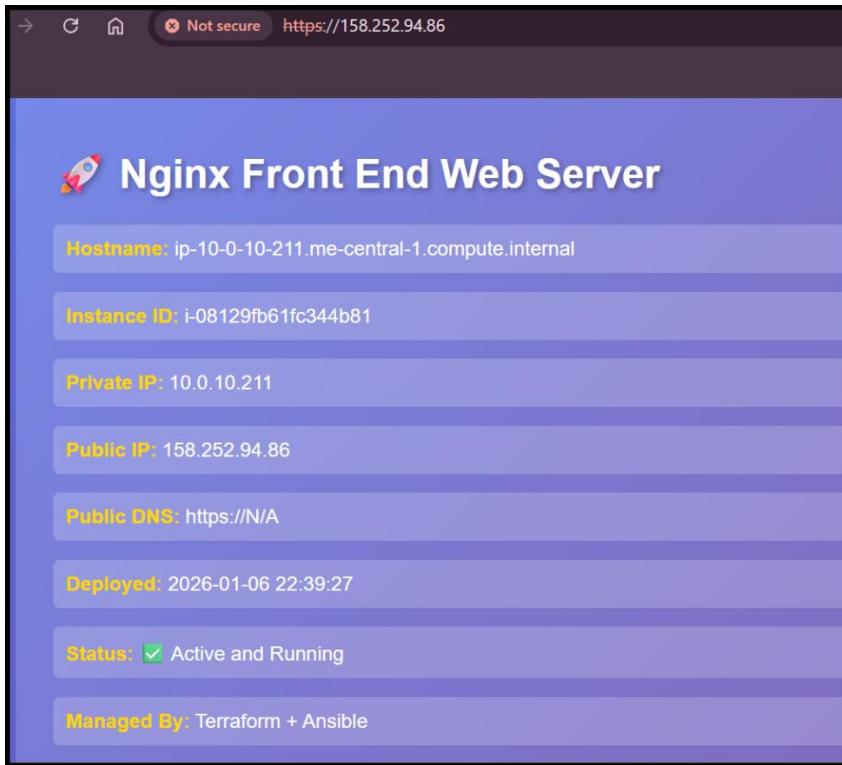
TASK [Restart nginx] ****
changed: [158.252.94.86]

TASK [Start and enable php-fpm] ****
changed: [158.252.94.86]

PLAY RECAP ****
158.252.94.86 : ok=17  changed=3    unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

@Musfira-0514 → /workspaces/terraform_machine (main) $
```

PHP application accessed over HTTPS



Task 8 – Docker & Docker Compose Installation

Previous infrastructure destroyed

```

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

Changes to Outputs:
  - webserver_public_ips = [
      - "158.252.94.86",
    ] -> null

module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-0eab18b0a49058ff4]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-webserver[0].aws_instance.myapp-server: Destroying... [id=i-08129fb61fc344b81]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id=igw-0239a2c2075c24d1d]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-08129fb61fc344b81, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0239a2c2075c24d1d, 00m10s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-08129fb61fc344b81, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0239a2c2075c24d1d, 00m20s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-08129fb61fc344b81, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0239a2c2075c24d1d, 00m30s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-08129fb61fc344b81, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0239a2c2075c24d1d, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 48s
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-08129fb61fc344b81, 00m50s elapsed]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [idsubnet-0835f25fdcef512a8]
module.myapp-webserver[0].aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-0]
module.myapp-webserver[0].aws_security_group.web_sg: Destroying... [id=sg-0a92e979754c7e311]
module.myapp-webserver[0].aws_key_pair.ssh-key: Destruction complete after 1s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 1s
module.myapp-webserver[0].aws_security_group.web_sg: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id=vpc-05a8e9baba768baa]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform apply for Docker-enabled instance

```

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

Changes to Outputs:
  + webserver_public_ips = [
      + (known after apply),
    ]
module.myapp-webserver[0].aws_key_pair.ssh-key: Creating...
aws_vpc.myapp_vpc: Creating...
module.myapp-webserver[0].aws_key_pair.ssh-key: Creation complete after 1s [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Creation complete after 2s [id=vpc-0cc03340f1b56f94e]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creating...
module.myapp-subnet.aws_subnet.myapp_subnet_1: Creating...
module.myapp-webserver[0].aws_security_group.web_sg: Creating...
module.myapp-subnet.aws_internet_gateway.myapp_igw: Creation complete after 0s [id=igw-0e2fe19596e38229c]
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-0cfe43893182b5a04]
module.myapp-webserver[0].aws_security_group.web_sg: Creation complete after 3s [id=sg-03ffffdd4a463be72a]
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-08d8ad2a0b51a1e05]
module.myapp-webserver[0].aws_instance.myapp-server: Creating...
module.myapp-webserver[0].aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Creation complete after 13s [id=i-0801c6dc307343719]

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

Outputs:

webserver_public_ips = [
  "51.112.54.21",
]
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform output showing new IP

```

● @Musfira-0514 → /workspaces/terraform_machine (main) $ terraform output
  webserver_public_ips = [
    "51.112.54.21",
  ]
● @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Inventory created for Docker servers

```

  ● @Musfira-0514 → /workspaces/terraform_machine (main) $ cat hosts
  [nginx]
  51.112.54.21

  [nginx:vars]
  ansible_ssh_private_key_file=~/ssh/id_ed25519
  ansible_user=ec2-user
  ● @Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Docker installation playbook

```

GNU nano 7.2                                         my-playbook.yaml *

---
- name: Configure Docker
  hosts: all
  become: true
  tasks:
    - name: install docker and update cache
      yum:
        name: docker
        state: present
        update_cache: yes

- name: Install Docker Compose
  hosts: all
  become: true
  gather_facts: true
  tasks:
    - name: create docker cli-plugins directory
      file:
        path: /usr/local/lib/docker/cli-plugins
        state: directory
        mode: '0755'

    - name: install docker-compose
      get_url:
        url: https://github.com/docker/compose/releases/latest/download/docker-compose-linux-{{ lookup('pipe', 'uname -m') }}
      dest: /usr/local/lib/docker/cli-plugins/docker-compose
      mode: +x

    - name: View architecture of the system
      debug:
        msg: "System architecture of {{ inventory_hostname }} is {{ ansible_facts['architecture'] }}"

- name: View architecture of the system
  debug:
    msg: "System architecture of {{ inventory_hostname }} is {{ lookup('pipe', 'uname -m') }}"

- name: Alternate method to view architecture of the system
  debug:
    msg: "System architecture of {{ inventory_hostname }} is {{ lookup('pipe', 'uname -m') }}"

- name: restart docker service
  service:
    name: docker
    state: restarted


```

Successful Docker installation

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml
ok: [51.112.54.21]

TASK [install docker and update cache] ****
changed: [51.112.54.21]

PLAY [Install Docker Compose] ****

TASK [Gathering Facts] ****
ok: [51.112.54.21]

TASK [create docker cli-plugins directory] ****
changed: [51.112.54.21]

TASK [install docker-compose] ****
changed: [51.112.54.21]

TASK [View architecture of the system] ****
ok: [51.112.54.21] => {
    "msg": "System architecture of 51.112.54.21 is x86_64"
}

TASK [Alternate method to view architecture of the system] ****
ok: [51.112.54.21] => {
    "msg": "System architecture of 51.112.54.21 is x86_64"
}

TASK [restart docker service] ****
changed: [51.112.54.21]

PLAY RECAP ****
51.112.54.21 : ok=8    changed=4    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

@Musfira-0514 → /workspaces/terraform_machine (main) $

```

Docker containers verified on remote host

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ ssh ec2-user@51.112.54.21 -i ~/.ssh/id_ed25519
#_
`__ _###_          Amazon Linux 2023
~~ \###\_
~~ \###_
~~ \#_
~~ \#_   https://aws.amazon.com/linux/amazon-linux-2023
~~ V~,_>
~~ /
~~ /_
~~ /m'

Last login: Tue Jan  6 22:57:26 2026 from 4.240.18.228
[ec2-user@ip-10-0-10-32 ~]$ sudo docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
[ec2-user@ip-10-0-10-32 ~]$ 

```

Task 9 – Gitea Deployment with Docker Compose

User added to Docker group via Ansible

```
GNU nano 7.2
- name: Adding user to docker group
  hosts: all
  become: true
  vars_files:
    - project-vars.yaml
  tasks:
    - name: add user to docker group
      user:
        name: "{{ normal_user }}"
        groups: docker
        append: yes

    - name: reconnect to apply group changes
      meta: reset_connection

    - name: verify docker access
      command: docker ps
      register: docker_ps
      changed_when: false

    - name: display docker ps output
      debug:
        var: docker_ps.stdout

    - name: fail if docker is not accessible
      fail:
        msg: "Docker is not accessible on this host"
      when: docker_ps.rc != 0
```

Project variables file configuration

```
● @Musfira-0514 ➔ /workspaces/terraform_machine (main) $ cat project-vars.yaml
normal_user: ec2-user
docker_compose_file_location: /workspaces/terraform_machine
```

Docker Compose deployment playbook

```
GNU nano 7.2
var: docker_ps.stdout
my-playbook.yaml *
- name: fail if docker is not accessible
  fail:
    msg: "Docker is not accessible on this host"
  when: docker_ps.rc != 0
- name: Deploy Docker Containers
  hosts: all
  become: true
  user: "{{ normal_user }}"
  vars_files:
    - project-vars.yaml
  tasks:
    - name: check if docker-compose file exists
      stat:
        path: /home/{{ normal_user }}/compose.yaml
      register: compose_file

    - name: copy docker-compose file
      copy:
        src: "{{ docker_compose_file_location }}/compose.yaml"
        dest: /home/{{ normal_user }}/compose.yaml
        mode: '0644'
      when: not compose_file.stat.exists

    - name: deploy containers using docker-compose
      command: docker compose up -d
      register: compose_result
      changed_when: "'Creating' in compose_result.stdout or 'Recreating' in compose_result.stdout"
```

Docker Compose YAML file

```

GNU nano 7.2
services:
  gitea:
    image: gitea/gitea:latest
    container_name: gitea
    environment:
      - DB_TYPE=postgres
      - DB_HOST=db:5432
      - DB_NAME=gitea
      - DB_USER=gitea
      - DB_PASSWORD=gitea
    restart: always
    volumes:
      - gitea:/data
    ports:
      - 3000:3000
    extra_hosts:
      - "www.jenkins.com:host-gateway"
  networks:
    - webnet

  db:
    image: postgres:alpine
    container_name: gitea_db
    environment:
      - POSTGRES_USER=gitea
      - POSTGRES_PASSWORD=gitea
      - POSTGRES_DB=gitea
    restart: always
    volumes:
      - gitea_postgres:/var/lib/postgresql/data
    volumes:
      - gitea_postgres:/var/lib/postgresql/data
    expose:
      - 5432
    networks:
      - webnet

  volumes:
    gitea_postgres:
      name: gitea_postgres
    gitea:
      name: gitea

  networks:
    webnet:
      name: webnet

```

Successful Gitea deployment

```

@Musfir-0514 ~/workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml
changed: [51.112.54.21]

TASK [reconnect to apply group changes] ****
TASK [verify docker access] ****
ok: [51.112.54.21]

TASK [display docker ps output] ****
ok: [51.112.54.21] => {
    "docker_ps_stdout": "CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES"
}

TASK [fail if docker is not accessible] ****
skipping: [51.112.54.21]

PLAY [Deploy Docker Containers] ****

TASK [Gathering Facts] ****
ok: [51.112.54.21]

TASK [check if docker-compose file exists] ****
ok: [51.112.54.21]

TASK [copy docker-compose file] ****
changed: [51.112.54.21]

TASK [deploy containers using docker-compose] ****
ok: [51.112.54.21]

PLAY RECAP ****
51.112.54.21 : ok=16    changed=3    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

@Musfir-0514 ~/workspaces/terraform_machine (main) $ 

```

Security group updated for port 3000

```
GNU nano 7.2
resource "aws_security_group" "web_sg" {
  vpc_id      = var.vpc_id
  name        = "${var.env_prefix}-web-sg-${var.instance_suffix}"
  description = "Security group for web server allowing HTTP, HTTPS and SSH"

  ingress {
    from_port  = 22
    to_port    = 22
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
  ingress {
    from_port  = 443
    to_port    = 443
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
  ingress {
    from_port  = 80
    to_port    = 80
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
  ingress {
    from_port  = 3000
    to_port    = 3000
    protocol   = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

Terraform applied security group changes

```
@Musfira-0514 ~/workspaces/terraform_machine (main) $ terraform apply -auto-approve
      + {
      +   + cidr_blocks  = [
      +     + "0.0.0.0/0",
      +   ]
      +   + from_port    = 80
      +   + ipv6_cidr_blocks = []
      +   + prefix_list_ids = []
      +   + protocol     = "tcp"
      +   + security_groups = []
      +   + self         = false
      +   + to_port      = 80
      + },
      ]
    name          = "dev-web-sg-0"
    tags          = {
      "Name" = "dev-default-sg"
    }
  # (9 unchanged attributes hidden)
}

Plan: 0 to add, 1 to change, 0 to destroy.
module.myapp-webserver[0].aws_security_group.web_sg: Modifying... [id=sg-03ffffdd4a463be72a]
module.myapp-webserver[0].aws_security_group.web_sg: Modifications complete after 1s [id=sg-03ffffdd4a463be72a]

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

Outputs:
webserver_public_ips = [
  "51.112.54.21",
]
@Musfira-0514 ~/workspaces/terraform_machine (main) $
```

Gitea web interface accessed in browser

The screenshot shows the Gitea configuration interface. It includes fields for Repository Root Path (set to /data/git/repositories), Git LFS Root Path (set to /data/git/lfs), Run As Username (set to git), Server Domain (set to 51.112.54.21), SSH Server Port (set to 22), Gitea HTTP Listen Port (set to 3000), Gitea Base URL (set to http://51.112.54.21:3000/), and Log Path (set to /data/gitea/log). There is also an optional checkbox for 'Enable Update Checker'. Below the configuration fields, a note states: 'These configuration options will be written into: /data/gitea/conf/app.ini'. A blue 'Install Gitea' button is at the bottom.

Task 10 – Automating Ansible Using Terraform

Terraform null_resource configuration

```
GNU nano 7.2
my_ip = local.my_ip
vpc_id = aws_vpc.myapp_vpc.id
subnet_id = module.myapp-subnet.subnet.id

# Loop count
count          = 1
# Use count.index to differentiate instances
instance_suffix = count.index
}

resource "null_resource" "configure_server" {
  triggers = {
    webserver_public_ips_for_ansible = join(
      ",",
      [for i in module.myapp-webserver : i.aws_instance.public_ip]
    )
  }

  depends_on = [module.myapp-webserver]

  provisioner "local-exec" {
    command = <<-EOT
      ansible-playbook \
      -i ${self.triggers.webserver_public_ips_for_ansible}, \
      --private-key "${var.private_key}" \
      --user ec2-user \
      my-playbook.yaml
    EOT
  }
}
```

Terraform destroy before automation

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ terraform destroy -auto-approve

Changes to Outputs:
- webserver_public_ips = [
  - "51.112.54.21",
] → null

module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-0cfec43893182b5a04]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-webserver[0].aws_instance.myapp-server: Destroying... [id=i-0001c6dc307343719]
module.myapp-webserver[0].aws_instance.myapp-server: Destroying... [id:igw-0e2fe19596e38229c]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id:igw-0e2fe19596e38229c, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id:igw-0e2fe19596e38229c, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m50s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id:igw-0e2fe19596e38229c, 00m50s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 58s
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0001c6dc307343719, 01m00s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Destruction complete after 1m1s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id:subnet-00d8ad2a0b51a1e05]
module.myapp-webserver[0].aws_key_pair.ssh-key: Destroying... [id:dev-serverkey-0]
module.myapp-webserver[0].aws_security_group.web_sg: Destroying... [id:sg-03ffffdd4a463be72a]
module.myapp-webserver[0].aws_key_pair.ssh-key: Destruction complete after 0s
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 1s
module.myapp-webserver[0].aws_security_group.web_sg: Destruction complete after 1s
aws_vpc.myapp_vpc: Destroying... [id:vpc-0cc03340f1b56f94e]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Terraform apply executing Ansible automatically

```

@Musfira-0514 → /workspaces/terraform_machine (main) $ terraform apply -auto-approve

null_resource.configure_server (local-exec): TASK [fail if docker is not accessible] ****
null_resource.configure_server (local-exec): skipping: [3.28.40.186]

null_resource.configure_server (local-exec): PLAY [Deploy Docker Containers] ****
null_resource.configure_server (local-exec): TASK [Gathering Facts] ****
null_resource.configure_server (local-exec): ok: [3.28.40.186]

null_resource.configure_server (local-exec): TASK [check if docker-compose file exists] ****
null_resource.configure_server: Still creating... [01m30s elapsed]
null_resource.configure_server (local-exec): ok: [3.28.40.186]

null_resource.configure_server (local-exec): TASK [copy docker-compose file] ****
null_resource.configure_server (local-exec): changed: [3.28.40.186]

null_resource.configure_server (local-exec): TASK [deploy containers using docker-compose] ****
null_resource.configure_server: Still creating... [01m40s elapsed]
null_resource.configure_server (local-exec): ok: [3.28.40.186]

null_resource.configure_server (local-exec): PLAY RECAP ****
null_resource.configure_server (local-exec): 3.28.40.186 : ok=16 changed=6 unreachable=0

null_resource.configure_server: Creation complete after 1m47s [id=5074850295984557789]

Apply complete! Resources: 8 added, 0 changed, 0 destroyed.

Outputs:

webserver_public_ips = [
  "3.28.40.186",
]

@Musfira-0514 → /workspaces/terraform_machine (main) $ 

```

Ansible wait-for-SSH task

```

GNU nano 7.2
- name: Wait for some time to ensure system readiness
  hosts: all
  tasks:
    - name: Wait 300 seconds for port 22 to become open and contain "OpenSSH"
      wait_for:
        port: 22
        host: "{{ inventory_hostname }}"
        delay: 10
        timeout: 300
        delegate_to: localhost

    ---

    - name: Configure Docker
      docker_container:
        name: docker
        image: docker:latest
        ports:
          - 2375:2375
        volumes:
          - /var/run/docker.sock:/var/run/docker.sock
        state: started

```

Successful Terraform apply after wait

```

@Musfira-0514 ~/workspaces/terraform_machine (main) $ terraform apply -auto-approve
null_resource.configure_server (local-exec): TASK [fail if docker is not accessible] ****
null_resource.configure_server (local-exec): skipping: [51.112.252.30]

null_resource.configure_server (local-exec): PLAY [Deploy Docker Containers] ****
null_resource.configure_server (local-exec): TASK [Gathering Facts] ****
null_resource.configure_server: Still creating... [03m0s elapsed]
null_resource.configure_server (local-exec): ok: [51.112.252.30]

null_resource.configure_server (local-exec): TASK [check if docker-compose file exists] ****
null_resource.configure_server (local-exec): ok: [51.112.252.30]

null_resource.configure_server (local-exec): TASK [copy docker-compose file] ****
null_resource.configure_server (local-exec): changed: [51.112.252.30]

null_resource.configure_server (local-exec): TASK [deploy containers using docker-compose] ****
null_resource.configure_server: Still creating... [03m0s elapsed]
null_resource.configure_server: Still creating... [03m50s elapsed]
null_resource.configure_server (local-exec): ok: [51.112.252.30]

null_resource.configure_server (local-exec): PLAY RECAP ****
null_resource.configure_server (local-exec): 51.112.252.30 : ok=18    changed=6    unreachable=0    failed=0    skipped=1
null_resource.configure_server: Creation complete after 3m51s [id=365751046382982292]

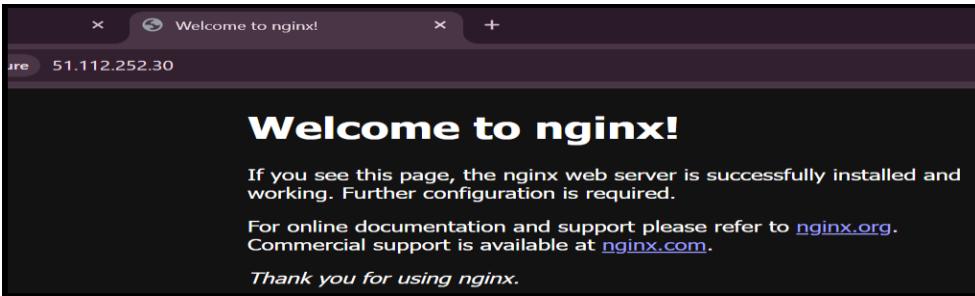
Apply complete! Resources: 8 added, 0 changed, 0 destroyed.

Outputs:

webserver_public_ips = [
  "51.112.252.30",
]
@Musfira-0514 ~/workspaces/terraform_machine (main) $ 

```

Application verified after automation



Task 11 – Dynamic Inventory Using aws_ec2 Plugin

Ansible configuration for aws_ec2 plugin

```

@Musfira-0514 ~/workspaces/terraform_machine (main) $ touch ansible.cfg
@Musfira-0514 ~/workspaces/terraform_machine (main) $ nano ansible.cfg
@Musfira-0514 ~/workspaces/terraform_machine (main) $ cat ansible.cfg
[defaults]
host_key_checking=False
interpreter_python = /usr/bin/python3
deprecation_warnings = False

enable_plugins = aws_ec2
private_key_file = ~/.ssh/id_ed25519

@Musfira-0514 ~/workspaces/terraform_machine (main) $ 

```

Dynamic inventory file creation

```

@Musfira-0514 ~/workspaces/terraform_machine (main) $ touch inventory_aws_ec2.yaml
ls -la inventory_aws_ec2.yaml
-rw-rw-rw- 1 codespace codespace 0 Jan 14 20:37 inventory_aws_ec2.yaml
@Musfira-0514 ~/workspaces/terraform_machine (main) $ 

```

Initial dynamic inventory configuration

```

GNU nano 7.2
---
plugin: aws_ec2
regions:
  - me-central-1

```

Terraform modules for dev and prod

```
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

  # Loop count
  count = 1
  # Use count.index to differentiate instances
  instance_suffix = count.index
}

resource "null_resource" "configure_server" {
  triggers = {
    webserver_public_ips_for_ansible = join(",", [
      for i in module.myapp-webserver : i.aws_instance.public_ip
    ])
  }
}

module "myapp-webserver-prod" {
  source = "./modules/webserver"
  env_prefix = "prod"
  instance_type = "t3.nano"
  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

  # Loop count
  count = 1
  # Use count.index to differentiate instances
  instance_suffix = count.index
}
```

Outputs file showing dev and prod IPs

```
GNU nano 7.2
output "webserver_public_ips" {
  value = [for i in module.myapp-webserver : i.aws_instance.public_ip]
}

output "prod-webserver_public_ips" {
  value = [for i in module.myapp-webserver-prod : i.aws_instance.public_ip]
}
```

Terraform output of dynamic instances

```
@Musfira-0514 ~/workspaces/terraform_machine (main) $ terraform apply -auto-approve
+ "Name" = "prod-default-sg"
}
+ tags_all = {
  + "Name" = "prod-default-sg"
}
+ vpc_id = "vpc-0042969d46f2df6cc"
}

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

Changes to Outputs:
+ prod-webserver_public_ips = [
  + (known after apply),
]

module.myapp-webserver-prod[0].aws_key_pair.ssh-key: Creating...
module.myapp-webserver-prod[0].aws_security_group.web_sg: Creating...
module.myapp-webserver-prod[0].aws_key_pair.ssh-key: Creation complete after 1s [id=prod-serverkey-0]
module.myapp-webserver-prod[0].aws_security_group.web_sg: Creation complete after 4s [id=sg-00d4fecbb4c664009]
module.myapp-webserver-prod[0].aws_instance.myapp-server: Creating...
module.myapp-webserver-prod[0].aws_instance.myapp-server: Still creating... [00m10s elapsed]
module.myapp-webserver-prod[0].aws_instance.myapp-server: Creation complete after 12s [id=i-08b376ab462d4c097]

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

Outputs:

prod-webserver_public_ips = [
  "40.172.232.236",
]
webserver_public_ips = [
  "51.112.252.30",
]
```

Boto library installation

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ $(which python) -m pip install boto3 botocore
Collecting boto3
  Downloading boto3-1.42.28-py3-none-any.whl.metadata (6.8 kB)
Collecting botocore
  Downloading botocore-1.42.28-py3-none-any.whl.metadata (5.9 kB)
Collecting jmespath<2.0.0,>=0.7.1 (from boto3)
  Downloading jmespath-1.0.1-py3-none-any.whl.metadata (7.6 kB)
Collecting s3transfer<0.17.0,>=0.16.0 (from boto3)
  Downloading s3transfer-0.16.0-py3-none-any.whl.metadata (1.7 kB)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /home/codespace/.local/lib/python3.12/site-packages (from botocore) (2.9.0.post0)
Requirement already satisfied: urllib3!=2.2.0,<3,>=1.25.4 in /home/codespace/.local/lib/python3.12/site-packages (from botocore) (2.5.0)
Requirement already satisfied: six>=1.5 in /home/codespace/.local/lib/python3.12/site-packages (from python-dateutil<3.0.0,>=2.1->botocore) (1.17.0)
Downloading boto3-1.42.28-py3-none-any.whl (140 kB)
  Downloading boto3-1.42.28-py3-none-any.whl (14.6 MB)
  14.6/14.6 MB 28.8 MB/s 0:00:00
  Downloading jmespath-1.0.1-py3-none-any.whl (20 kB)
  Downloading s3transfer-0.16.0-py3-none-any.whl (86 kB)
  Installing collected packages: jmespath, botocore, s3transfer, boto3
  Successfully installed boto3-1.42.28 botocore-1.42.28 jmespath-1.0.1 s3transfer-0.16.0
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Boto version verification

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ $(which python) -c "import boto3, botocore; print(boto3.__version__)"
1.42.28
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Dynamic inventory graph output

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-inventory -i inventory_aws_ec2.yaml --graph
@all:
  |--@ungrouped:
  |--@aws_ec2:
    | |--ip-10-0-10-84.me-central-1.compute.internal
    | |--ip-10-0-10-244.me-central-1.compute.internal
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Task 12 – Inventory Filtering & Grouping

Dynamic inventory grouped by tags

```
GNU nano 7.2
---
plugin: amazon.aws.aws_ec2
regions:
  - me-central-1

keyed_groups:
  - key: tags
    prefix: tag
    separator: "_"
```

Inventory graph showing tag-based groups

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ nano inventory_aws_ec2.yaml
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-inventory -i inventory_aws_ec2.yaml --graph

@all:
  |--@ungrouped:
  |--@aws_ec2:
    | |--ip-10-0-10-84.me-central-1.compute.internal
    | |--ip-10-0-10-244.me-central-1.compute.internal
    | |--tag_Name_dev_ec2_instance_0:
    |   |--ip-10-0-10-84.me-central-1.compute.internal
    | |--tag_Name_prod_ec2_instance_0:
    |   |--ip-10-0-10-244.me-central-1.compute.internal
○ @Musfira-0514 → /workspaces/terraform_machine (main) $
```

Inventory grouped by instance type

```
GNU nano 7.2
---
plugin: amazon.aws.aws_ec2
regions:
  - me-central-1

keyed_groups:
  - key: tags
    prefix: tag
    separator: "_"

  - key: instance_type
    prefix: instance_type
    separator: "_"
```

Full inventory graph view

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ nano inventory_aws_ec2.yaml
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-inventory -i inventory_aws_ec2.yaml --graph
@all:
  |--ungrouped:
  |--@aws_ec2:
  |  |--ip-10-0-10-84.me-central-1.compute.internal
  |  |--ip-10-0-10-244.me-central-1.compute.internal
  |--@tag_Name_dev_ec2_instance_0:
  |  |--ip-10-0-10-84.me-central-1.compute.internal
  |--@instance_type_t3_micro:
  |  |--ip-10-0-10-84.me-central-1.compute.internal
  |--@tag_Name_prod_ec2_instance_0:
  |  |--ip-10-0-10-244.me-central-1.compute.internal
  |--@instance_type_t3.nano:
  |  |--ip-10-0-10-244.me-central-1.compute.internal
```

Playbook execution on all hosts

```
GNU nano 7.2
- -
- name: Configure nginx web server
  hosts: all
  become: true
  tasks:
    - name: install nginx and update cache
      yum:
        name: nginx
        state: present
        update_cache: yes

    - name: install openssl
      yum:
        name: openssl
        state: present

    - name: start nginx server
      service:
        name: nginx
        state: started
        enabled: true

- name: Configure SSL certificates
  hosts: all
  become: true
  tasks:
    - name: Create SSL private directory
      file:
        path: /etc/ssl/private
        state: directory
```

Playbook execution on dev hosts

```
● @Musfira-0514 → /workspaces/terraform_machine (main) $ ansible-inventory -i inventory_aws_ec2.yaml --graph
@all:
  |--@ungrouped:
  |--@aws_ec2:
  |  |--public-ip-address
  |--@tag_Name_dev_ec2_instance_0:
  |  |--public-ip-address
  |--@instance_type_t3_micro:
  |  |--public-ip-address
  |--@tag_Name_prod_ec2_instance_0:
  |  |--public-ip-address
  |--@instance_type_t3.nano:
  |  |--public-ip-address
  ○ @Musfira-0514 → /workspaces/terraform_machine (main) $ █
```

Playbook execution on prod hosts

```
@Musfira-0514 ~/workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i inventory_aws_ec2.yaml -u ec2-user --private-key ~/.ssh/id_ed25519 -l "tag_Name_dev_%" my-playbook.yaml

TASK [Show current public IP] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com] => {
    "msg": "Public IP: 40.172.221.203"
}

TASK [Save public IP as fact] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Generate self-signed SSL certificate] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-203.me-central-1.compute.amazonaws.com : ok=18  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Playbook execution on t3.micro instances

```
@Musfira-0514 ~/workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i inventory_aws_ec2.yaml -u ec2-user --private-key ~/.ssh/id_ed25519 -l "tag_Name_prod_%" my.playbook.yaml

TASK [Show current public IP] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com] => {
    "msg": "Public IP: 40.172.221.121"
}

TASK [Save public IP as fact] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Generate self-signed SSL certificate] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-121.me-central-1.compute.amazonaws.com : ok=18  changed=1  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

Playbook execution on t3.nano instances

```

@Musfir-0514~/workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i inventory_aws_ec2.yaml -u ec2-user --private-key ~/.ssh/id_ed25519 -l instance_type_t3_micro my-playbook.yaml

TASK [Save public IP as fact] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Generate self-signed SSL certificate] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-203.me-central-1.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-203.me-central-1.compute.amazonaws.com : ok=18    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

@Musfir-0514~/workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i inventory_aws_ec2.yaml -u ec2-user --private-key ~/.ssh/id_ed25519 -l instance_type_t3.nano my-playbook.yaml

TASK [Show current public IP] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com] => {
    "msg": "Public IP: 40.172.221.121"
}

TASK [Save public IP as fact] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Generate self-signed SSL certificate] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-121.me-central-1.compute.amazonaws.com : ok=18    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

```

Default inventory configured in ansible.cfg

```

GNU nano 7.2
[defaults]
host_key_checking = False
interpreter_python = /usr/bin/python3
deprecation_warnings = False
enable_plugins = aws_ec2
private_key_file = ~/.ssh/id_ed25519
inventory = ./inventory_aws_ec2.yaml

```

Playbook execution without specifying inventory

```
@Musfira-0514 → /workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -u ec2-user --private-key ~/.ssh/id_ed25519 -i instance_type_t3 nano my-playbook.yaml

TASK [Show current public IP] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com] => {
  "msg": "Public IP: 40.172.221.121"
}

TASK [Save public IP as fact] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Generate self-signed SSL certificate] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY [Deploy Nginx website and configuration files] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-121.me-central-1.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-121.me-central-1.compute.amazonaws.com : ok=18    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Task 13 – Ansible Roles Implementation

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

```
}

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

  # Loop count
  count           = 1
  # Use count.index to differentiate instances
  instance_suffix = count.index
}

resource "null_resource" "configure_server" {
  triggers = [
    public_ip = module.myapp-webserver[0].aws_instance.public_ip
  ]

  depends_on = [module.myapp-webserver]

  provisioner "local-exec" {
    environment = {
      ANSIBLE_HOST_KEY_CHECKING = "False"
    }
    # We use a single line here to prevent Windows \r errors
    command = "sleep 30 && echo '[docker_servers]' > hosts && echo '${self.triggers.public_ip}' >> hosts && echo '[docker_servers]' >> hosts"
  }
}
```

```
@Musfira-0514 → /workspaces/terraform_machine (main) $ mkdir -p ansible && cd ansible
mkdir inventory roles
touch ansible.cfg my-playbook.yaml
ls -R
.:
ansible.cfg  inventory  my-playbook.yaml  roles

./inventory:

./roles:
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
GNU nano 7.2                                         ansible.cfg *
```

[defaults]
host_key_checking = False
interpreter_python = /usr/bin/python3

```
GNU nano 7.2                                         inventory/hosts *
```

[nginx]
40.172.221.203

[nginx:vars]
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_user=ec2-user

```
@Musfira-0514 →/workspaces/terraform_machine/ansible (main) $ ls -R roles  
main.yml  
  
roles/ssl/tasks:  
main.yml  
  
roles/ssl/templates:  
  
roles/ssl/tests:  
inventory test.yml  
  
roles/ssl/vars:  
main.yml  
  
roles/webapp:  
README.md  defaults  files  handlers  meta  tasks  templates  tests  vars  
roles/webapp/defaults:  
main.yml  
  
roles/webapp/files:  
  
roles/webapp/handlers:  
main.yml  
  
roles/webapp/meta:  
main.yml  
  
roles/webapp/tasks:  
main.yml  
  
roles/webapp/templates:  
  
roles/webapp/tests:  
inventory test.yml  
  
roles/webapp/vars:  
main.yml
```

```
GNU nano 7.2                                         roles/nginx/handlers/main.yml *
```

#SPDX-License-Identifier: MIT-0

handlers file for nginx
- name: Restart nginx
 service:
 name: nginx
 state: restarted

```
GNU nano 7.2                                         roles/nginx/tasks/main.yml *
#SPDX-License-Identifier: MIT-0
---
# tasks file for nginx
- name: Install nginx
  yum:
    name: nginx
    state: present
    update_cache: yes
  notify: Restart nginx

- name: Install openssl
  yum:
    name: openssl
    state: present

- name: Start and enable nginx
  service:
    name: nginx
    state: started
    enabled: true
```

```
GNU nano 7.2                                         my-playbook.yaml *
---
- name: Deploy NGINX Web Stack with SSL and PHP
  hosts: nginx
  become: true
  roles:
    - nginx
```

```
@Musfiq-0514~/workspaces/terraform_machine/ansible (main) $ ansible-playbook -i inventory/hosts my-playbook.yaml
```

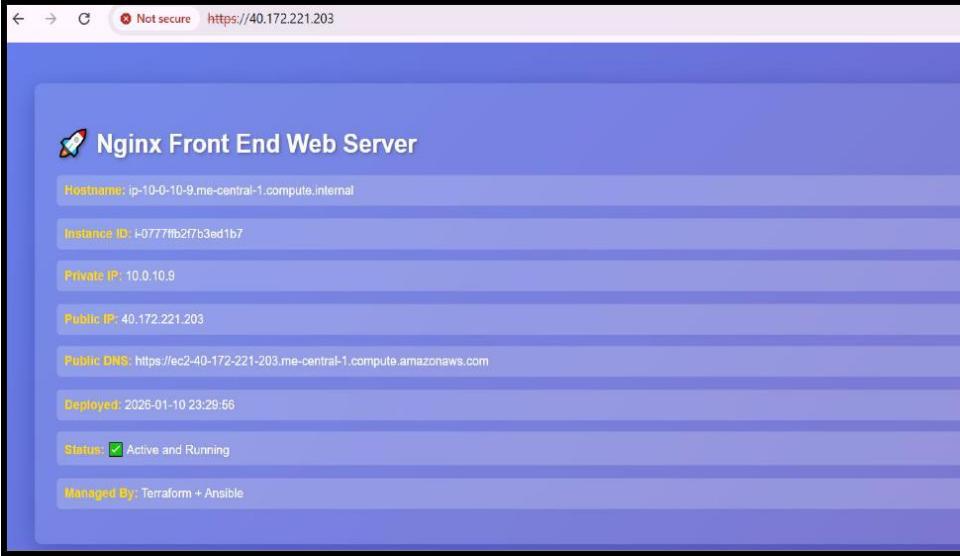
```
PLAY [Deploy NGINX Web Stack with SSL and PHP] ****
TASK [Gathering Facts] ****
ok: [40.172.221.203]

TASK [nginx : Install nginx] ****
ok: [40.172.221.203]

TASK [nginx : Install openssl] ****
ok: [40.172.221.203]

TASK [nginx : Start and enable nginx] ****
ok: [40.172.221.203]

PLAY RECAP ****
40.172.221.203 : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

GNU nano 7.2 roles/ssl/defaults/main.yml *

```
#SPDX-License-Identifier: MIT-0
---
# defaults file for ssl
imdsv2_token_ttl: "3600"
ssl_days_valid: 365
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

GNU nano 7.2 roles/ssl/tasks/main.yml *

```
#SPDX-License-Identifier: MIT-0
---
# tasks file for ssl
- name: Create SSL private directory
  file:
    path: /etc/ssl/private
    state: directory
    mode: '0700'

- name: Create SSL certs directory
  file:
    path: /etc/ssl/certs
    state: directory
    mode: '0755'

- name: Get IMDSv2 token
  uri:
    url: http://169.254.169.254/latest/api/token
    method: PUT
    headers:
      X-aws-ec2-metadata-token-ttl-seconds: "{{ imdsv2_token_ttl }}"
    return_content: yes
  register: imds_token

- name: Get public IP
  uri:
    url: http://169.254.169.254/latest/meta-data/public-ipv4
    headers:
      X-aws-ec2-metadata-token: "{{ imds_token.content }}"
    return_content: yes
  register: public_ip

- name: Save public IP as fact
  set_fact:
    server_public_ip: "{{ public_ip.content }}"
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

GNU nano 7.2 roles/webapp/defaults/main.yml *

```
#SPDX-License-Identifier: MIT-0
---
# defaults file for webapp
nginx_user: nginx
nginx_worker_processes: auto
nginx_worker_connections: 1024
nginx_error_log_level: notice
# Webapp settings
web_root: /usr/share/nginx/html
web_index_file: index.php
```

```

GNU nano 7.2                                         roles/webapp/files/index.php *
<?php
// Get hostname
$hostname = gethostname();

// Deployment date
$deployed_date = date("Y-m-d H:i:s");

// Metadata base URL
$metadata_base = "http://169.254.169.254/latest/";

// Function to get IMDSv2 token
function getImdsv2Token() {
    $ch = curl_init("http://169.254.169.254/latest/api/token");
    curl_setopt_array($ch, [
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_CUSTOMREQUEST => "PUT",
        CURLOPT_HTTPHEADER => [
            "X-aws-ec2-metadata-token-ttl-seconds: 21600"
        ],
        CURLOPT_TIMEOUT => 2
    ]);

    $token = curl_exec($ch);
    curl_close($ch);

    return $token ?: null;
}

// Function to fetch metadata using token
function getMetadata($path, $token) {
    $url = "http://169.254.169.254/latest/meta-data/" . $path;

    $ch = curl_init($url);
    curl_setopt_array($ch, [

```

```

GNU nano 7.2                                         roles/webapp/handlers/main.yml *
#SPDX-License-Identifier: MIT-0
...
# handlers file for webapp
- name: Restart nginx
  service:
    name: nginx
    state: restarted

- name: Restart php-fpm
  service:
    name: php-fpm
    state: restarted

```

```

GNU nano 7.2                                         roles/webapp/templates/nginx.conf.j2 *
user {{ nginx_user }};
worker_processes {{ nginx_worker_processes }};
error_log /var/log/nginx/error.log {{ nginx_error_log_level }};
pid /run/nginx.pid;

events {
    worker_connections {{ nginx_worker_connections }};
}

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 {{ server_public_ip }};

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

```

File menu: Help, Undo, Redo, Cut, Paste, Execute, Location, Help, Exit, Save, Mail

```

GNU nano 7.2                                         roles/webapp/tasks/main.yml *
#SPDX-License-Identifier: MIT-0
---
# tasks file for webapp
- name: Install PHP packages
  yum:
    name:
      - php-fpm
      - php-curl
    state: present
  notify: Restart php-fpm

- name: Copy PHP website
  copy:
    src: index.php
    dest: "{{ web_root }}/{{ web_index_file }}"
    owner: nginx
    group: nginx
    mode: '0644'
  notify: Restart nginx

- name: Deploy nginx config
  template:
    src: nginx.conf.j2
    dest: /etc/nginx/nginx.conf
  notify: Restart nginx

- name: Start and enable php-fpm
  service:
    name: php-fpm
    state: started
    enabled: true

```

```

GNU nano 7.2                                         my-playbook.yaml *

---
- name: Deploy NGINX Web Stack with SSL and PHP
  hosts: nginx
  become: true
  roles:
    - nginx
    - ssl
    - webapp

@Musfira-0514 → /workspaces/terraform_machine/ansible (main) $ ansible-playbook -i inventory/hosts my-playbook.yaml

TASK [ssl : Create SSL private directory] ****
ok: [40.172.221.203]

TASK [ssl : Create SSL certs directory] ****
ok: [40.172.221.203]

TASK [ssl : Get IMDSv2 token] ****
ok: [40.172.221.203]

TASK [ssl : Get public IP] ****
ok: [40.172.221.203]

TASK [ssl : Save public IP as fact] ****
ok: [40.172.221.203]

TASK [ssl : Generate self-signed certificate] ****
ok: [40.172.221.203]

TASK [webapp : Install PHP packages] ****
ok: [40.172.221.203]

TASK [webapp : Copy PHP website] ****
ok: [40.172.221.203]

TASK [webapp : Deploy nginx config] ****
changed: [40.172.221.203]

TASK [webapp : Start and enable php-fpm] ****
ok: [40.172.221.203]

RUNNING HANDLER [webapp : Restart nginx] ****
changed: [40.172.221.203]

PLAY RECAP ****
40.172.221.203 : ok=15  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

→ ⚡ Not secure https://40.172.221.203

  Nginx Front End Web Server

  Hostname: ip-10-0-10-9.me-central-1.compute.internal
  Instance ID: i-0777fb2f7b3ed1b7
  Private IP: 10.0.10.9
  Public IP: 40.172.221.203
  Public DNS: https://ec2-40-172-221-203.me-central-1.compute.amazonaws.com
  Deployed: 2026-01-10 23:41:41
  Status:  Active and Running
  Managed By: Terraform + Ansible

```

Cleanup

cleanup_terraform_destroy

```

ok: [40.172.221.203]

TASK [ssl : Generate self-signed certificate] ****
ok: [40.172.221.203]

TASK [webapp : Install PHP packages] ****
ok: [40.172.221.203]

TASK [webapp : Copy PHP website] ****
ok: [40.172.221.203]

TASK [webapp : Deploy nginx config] ****
changed: [40.172.221.203]

TASK [webapp : Start and enable php-fpm] ****
ok: [40.172.221.203]

RUNNING HANDLER [webapp : Restart nginx] ****
changed: [40.172.221.203]

PLAY RECAP ****
40.172.221.203 : ok=15  changed=2    unreachable=0   failed=0   skipped=0   rescued=0   ignored=0

@Musfira-0514 → /workspaces/terraform_machine/ansible (main) $ terraform destroy -auto-approve

No changes. No objects need to be destroyed.

Either you have not created any objects yet or the existing objects were already deleted outside of Terraform.

Destroy completed. Resources: 0 destroyed.

```

cleanup_tfstate

```

@Musfira-0514 → /workspaces/terraform_machine/ansible (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.14.3",
  "serial": 1,
  "lineage": "a12d12a8-16fa-b810-7175-d93be1f05675",
  "outputs": {},
  "resources": [],
  "check_results": null
}

```

cleanup_aws_console

