

Name: Sadaf Riaz

Roll no: 2023-BSE-077

Section: BSE5-B

Lab 14 – Terraform + Ansible

Task 0 – Lab Setup (Codespace, GH CLI & AWS Access)

task0_codespace_open.png

GitHub Repository Opened in Codespace Environment

```
@SadafRiaz-077 →/workspaces/cc_sadafriaz_077_lab-14 (main) $
```

task0_env_check.png

Verification of Preinstalled Tools (AWS CLI, Terraform, Ansible, GH CLI)

```
"Install Ansible core ..."  
@SadafRiaz-077 →/tmp $ ansible --version  
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
```

task0_aws_config.png

AWS CLI Configured with Access Keys

```
@SadafRiaz-077 →/workspaces/cc_sadafriaz_077_lab-14 (main) $ aws configure  
AWS Access Key ID [None]: AKIAV4QTXR5S75HNE000  
AWS Secret Access Key [None]: 7deGEgSEuCUSkT2xW91QW5B7v0+0SkEebJSmZc66  
Default region name [None]: me-central-1  
Default output format [None]: json  
@SadafRiaz-077 →/workspaces/cc_sadafriaz_077_lab-14 (main) $ aws sts get-caller-identity  
{  
    "UserId": "AIDAV4QTXR5SUMEPNC2FG",  
    "Account": "404842057573",  
    "Arn": "arn:aws:iam::404842057573:user/lab14"  
}  
@SadafRiaz-077 →/workspaces/cc_sadafriaz_077_lab-14 (main) $
```

Task 1 – Clone Repository & Initial Terraform Deployment

task1_repo_cloned.png

```
● @SadafRiaz-077 →/workspaces $ cd /workspaces
git clone --branch main https://github.com/WaqasSaleem97/terraform_machine.git
Cloning into 'terraform_machine'...
remote: Enumerating objects: 30, done.
remote: Counting objects: 100% (30/30), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 30 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (30/30), 7.03 KiB | 7.03 MiB/s, done.
Resolving deltas: 100% (5/5), done.
● @SadafRiaz-077 →/workspaces $ cd terraform_machine
ls -la
total 40
drwxrwxrwx+ 4 codespace codespace 4096 Jan  6 15:47 .
drwxr-xrwx+ 6 codespace root      4096 Jan  6 15:47 ..
drwxrwxrwx+ 7 codespace codespace 4096 Jan  6 15:47 .git
-rw-rw-rw-  1 codespace codespace  407 Jan  6 15:47 .gitignore
-rw-rw-rw-  1 codespace codespace   20 Jan  6 15:47 README.md
-rw-rw-rw-  1 codespace codespace  128 Jan  6 15:47 locals.tf
-rw-rw-rw-  1 codespace codespace  973 Jan  6 15:47 main.tf
drwxrwxrwx+ 4 codespace codespace 4096 Jan  6 15:47 modules
-rw-rw-rw-  1 codespace codespace  108 Jan  6 15:47 outputs.tf
-rw-rw-rw-  1 codespace codespace  202 Jan  6 15:47 variables.tf
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

Terraform and Ansible Repository Successfully Cloned

task1_ssh_keygen_before.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ls ~/.ssh
known_hosts
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

SSH Directory State Before Key Generation

task1_ssh_keygen.png

```
@SadafRiaz-077 →/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:
SHA256:C5lg5ugcbjxiJHqvbiW0reVLcoUZT06jetjCpxUNiwU codespace@codespaces-836f0e
The key's randomart image is:
++-[ED25519 256]--+
| E .
| .+
| oB
| +*= o
| .+o=.= S
| +=oo=o . .
| ++@Bo .
| +**B.
| .*=.o.
+---[SHA256]----+
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

SSH Key Pair Generated Using ssh-keygen

task1_ssh_keygen_after.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ls -la ~/.ssh
total 24
drwx----- 2 codespace codespace 4096 Jan  6 15:49 .
drwxr-x--- 1 codespace codespace 4096 Jan  6 15:45 ..
-rw----- 1 codespace codespace  419 Jan  6 15:49 id_ed25519
-rw-r--r-- 1 codespace codespace 109 Jan  6 15:49 id_ed25519.pub
-rw-r--r-- 1 codespace codespace 142 Jan  6 15:45 known hosts
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

SSH Key Pair Successfully Created

task1_terraform_tfvars_created.png

terraform.tfvars File Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cd /workspaces/terraform_machine
touch terraform.tfvars
ls -la terraform.tfvars
-rw-rw-rw- 1 codespace codespace 0 Jan  6 15:52 terraform.tfvars
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task1_terraform_tfvars.png

terraform.tfvars Configured with Required Variables

```
@SadafRiaz-077 →/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"
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task1_terraform_init.png

Terraform Initialized Successfully

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform init
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!

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.
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task1_terraform_apply_2_instances.png

Terraform Apply Completed – Two EC2 Instances Created

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

Outputs:

```
webserver_public_ips = [
  "3.28.41.237",
  "3.28.182.84",
]
```

task1_terraform_output_ips.png

Terraform Output Displaying EC2 Public IP Addresses

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.41.237",
  "3.28.182.84",
]
```

Task 2 – Static Ansible Inventory with Two EC2 Instances

task2_ansible_install.png

Ansible Installed Successfully Using pipx

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible --version
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)
```

task2_terraform_output_ips.png

Retrieving EC2 Public IPs Using Terraform Output

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.41.237",
  "3.28.182.84",
]
```

task2_hosts_created.png

Static Ansible Inventory File Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cd /workspaces/terraform_machine  
touch hosts  
ls -la hosts  
-rw-rw-rw- 1 codespace codespace 0 Jan  6 16:10 hosts  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task2_ansible_ping_initial.png

Initial Ansible Ping Test Execution

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano hosts  
@SadafRiaz-077 →/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}  
  
3.28.41.237 | UNREACHABLE! => {  
    "changed": false,  
    "msg": "Task failed: Failed to connect to the host via ssh: Host key verification failed.",  
    "unreachable": true  
}  
3.28.182.84 | UNREACHABLE! => {  
    "changed": false,  
    "msg": "Task failed: Failed to connect to the host via ssh: Host key verification failed.",  
    "unreachable": true  
}
```

task2_hosts_with_common_args.png

Inventory Updated to Disable Strict Host Key Checking

```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat hosts  
3.28.41.237 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'  
3.28.182.84 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'  
█
```

task2_ansible_ping_success.png

Successful Ansible Ping to All EC2 Hosts

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano hosts  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible all -i hosts -m ping  
[WARNING]: Host '3.28.41.237' 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.  
3.28.41.237 | SUCCESS => {  
    "ansible_facts": {  
        "discovered_interpreter_python": "/usr/bin/python3.9"  
    },  
    "changed": false,  
    "ping": "pong"  
}  
[WARNING]: Host '3.28.182.84' 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.  
3.28.182.84 | SUCCESS => {  
    "ansible_facts": {  
        "discovered_interpreter_python": "/usr/bin/python3.9"  
    },  
    "changed": false,  
    "ping": "pong"  
}
```

Task 3 – Scale to Three Instances & Group-Based Inventory

task3_main_tf_count_3.png

Terraform Configuration Updated to Use Count = 3

```
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

  # Scale to 3 instances
  count        = 3
  instance_suffix = count.index
}
```

task3_terraform_apply_3_instances.png

Terraform Applied – Three EC2 Instances Running

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

```
Outputs:
```

```
webserver_public_ips = [
  "3.28.41.237",
  "3.28.182.84",
  "3.28.201.47",
]
```

```
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $
```

task3_terraform_output_3_ips.png

Terraform Output Showing Three EC2 Public Ips

```
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.41.237",
  "3.28.182.84",
  "3.28.201.47",
]
```

```
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $
```

task3_hosts_grouped.png

Ansible Inventory Organized Using Host Groups

```

@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ nano /workspaces/terraform_machine/hosts
@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ cat nano /workspaces/terraform_machine/hosts
cat: nano: No such file or directory
[ec2]
3.28.41.237
3.28.182.84

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

[droplet]
3.28.201.47

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

@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ []

```

task3_ansible_ec2_ping.png

Ansible Ping Test Executed on EC2 Group

```

@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ ansible ec2 -i hosts -m ping
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/python3.9', but future
discovery 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.
3.28.41.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.182.84' is using the discovered Python interpreter at '/usr/bin/python3.9', but future
discovery 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.
3.28.182.84 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ []

```

task3_ansible_single_ip_ping.png

Ansible Ping Test Executed on a Single EC2 Host

```

@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ ansible 3.28.41.237 -i hosts -m ping
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/python3.9', but future
discovery 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.
3.28.41.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
@SadafRiaz-077 ➔/workspaces/terraform_machine (main) $ []

```

task3_ansible_droplet_ping.png

Ansible Ping Test Executed on Droplet Group

```
[@SadafRiaz-077 ~] /workspaces/terraform_machine (main) $ ansible droplet -i hosts -m ping
[WARNING]: Host '3.28.201.47' is using the discovered Python interpreter at '/usr/bin/python3.9', 
ion of another Python interpreter could cause a different interpreter to be discovered. See https://
ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.201.47 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
```

task3_ansible_all_ping.png

Ansible Ping Test Executed on All Hosts

```
[@SadafRiaz-077 ~] /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '3.28.201.47' is using the discovered Python interpreter at '/usr/bin/pyt
ion of another Python interpreter could cause a different interpreter to be discovered. S
nsible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.201.47 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/pyt
ion of another Python interpreter could cause a different interpreter to be discovered. S
nsible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.41.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.182.84' is using the discovered Python interpreter at '/usr/bin/pyt
ion of another Python interpreter could cause a different interpreter to be discovered. S
nsible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.182.84 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[@SadafRiaz-077 ~] /workspaces/terraform_machine (main) $
```

Task 4 – Global ansible.cfg & First Nginx Playbook

task4_global_ansible_cfg.png

Global Ansible Configuration File Created

```
@SadafRiaz-077 ~] /workspaces/terraform_machine (main) $ vim ~/.ansible.cfg
[@SadafRiaz-077 ~] /workspaces/terraform_machine (main) $ cat ~/.ansible.cfg
[default]
host_key_checking = False
interpreter_python = /usr/bin/python3
```

task4_hosts_without_common_args.png

Inventory Updated After Removing SSH Common Arguments

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat hosts
[ec2]
3.28.41.237
3.28.182.84

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

[droplet]
3.28.201.47

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

Inventory Updated After Removing SSH Common Arguments

task4_ansible_ping_after_cfg.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/python3.9'.
           This is a security risk as it could cause a different interpreter to be discovered. See http://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.41.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.201.47' is using the discovered Python interpreter at '/usr/bin/python3.9'.
           This is a security risk as it could cause a different interpreter to be discovered. See http://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.201.47 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '3.28.182.84' is using the discovered Python interpreter at '/usr/bin/python3.9'.
           This is a security risk as it could cause a different interpreter to be discovered. See http://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.182.84 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

Ansible Ping Verified Using Global Configuration

task4_my_playbook_created.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ touch my-playbook.yaml  
ls -la my-playbook.yaml  
-rw-rw-rw- 1 codespace codespace 0 Jan  6 16:58 my-playbook.yaml  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ [ ]
```

Initial Nginx Ansible Playbook Created

task4_my_playbook_ec2.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano my-playbook.yaml  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml  
---  
- 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  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ [ ]
```

Nginx Playbook Configured for EC2 Group

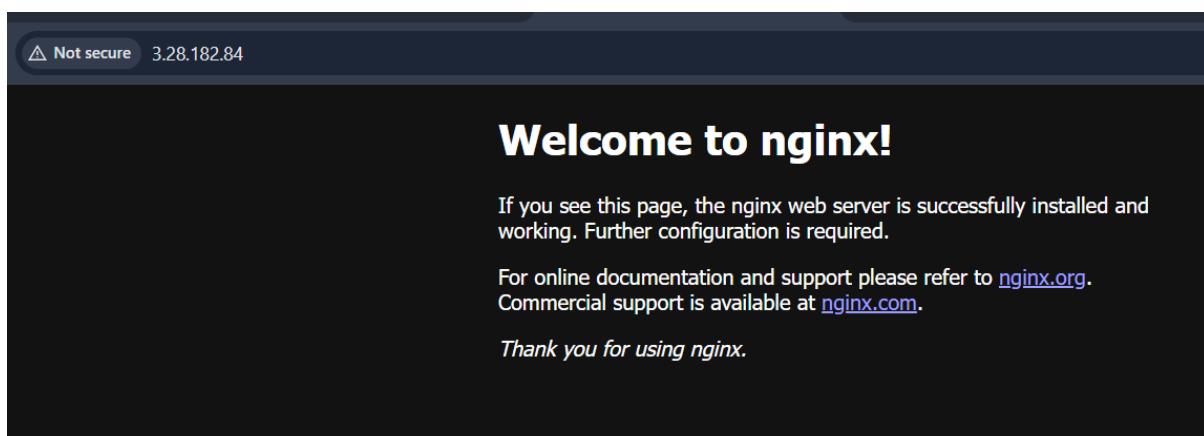
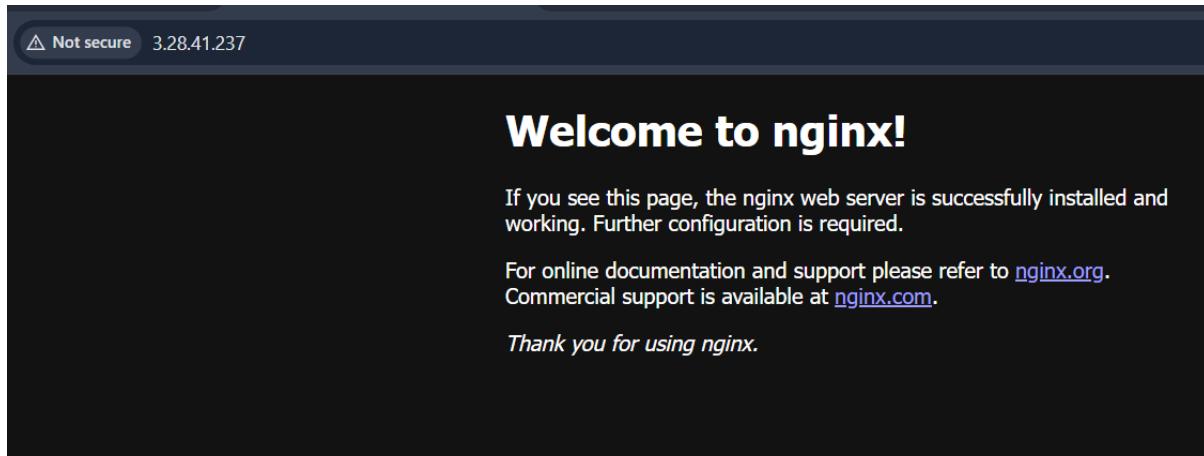
task4_ansible_play_ec2.png

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml  
PLAY [Configure nginx web server] *****  
  
TASK [Gathering Facts] *****  
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/python3.9', but  
ation of another Python interpreter could cause a different interpreter to be discovered. See https://do  
nsible-core/2.20/reference_appendices/interpreter_discovery.html for more information.  
ok: [3.28.41.237]  
[WARNING]: Host '3.28.182.84' is using the discovered Python interpreter at '/usr/bin/python3.9', but  
ation of another Python interpreter could cause a different interpreter to be discovered. See https://do  
nsible-core/2.20/reference_appendices/interpreter_discovery.html for more information.  
ok: [3.28.182.84]  
  
TASK [install nginx and update cache] *****  
changed: [3.28.182.84]  
changed: [3.28.41.237]  
  
TASK [start nginx server] *****  
changed: [3.28.182.84]  
changed: [3.28.41.237]  
  
PLAY RECAP *****  
3.28.182.84 : ok=3     changed=2     unreachable=0     failed=0     skipped=0     rescued=0  
3.28.41.237 : ok=3     changed=2     unreachable=0     failed=0     skipped=0     rescued=0  
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ [ ]
```

Nginx Installed Successfully on EC2 Instances

task4_nginx_browser_ec2.png

Nginx Default Page Verified on EC2 Instance



task4_my_playbook_droplet.png

Playbook Updated to Target Droplet Group

```
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $ vim my-playbook.yaml
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $ cat my-playbook.yaml
---
- 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
@SadafRiaz-077 ➔ /workspaces/terraform_machine (main) $
```

Playbook Updated to Target Droplet Group

task4_ansible_play_droplet.png

Nginx Installed Successfully on Droplet Instance

```
state: started
@sadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.201.47' 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.
ok: [3.28.201.47]

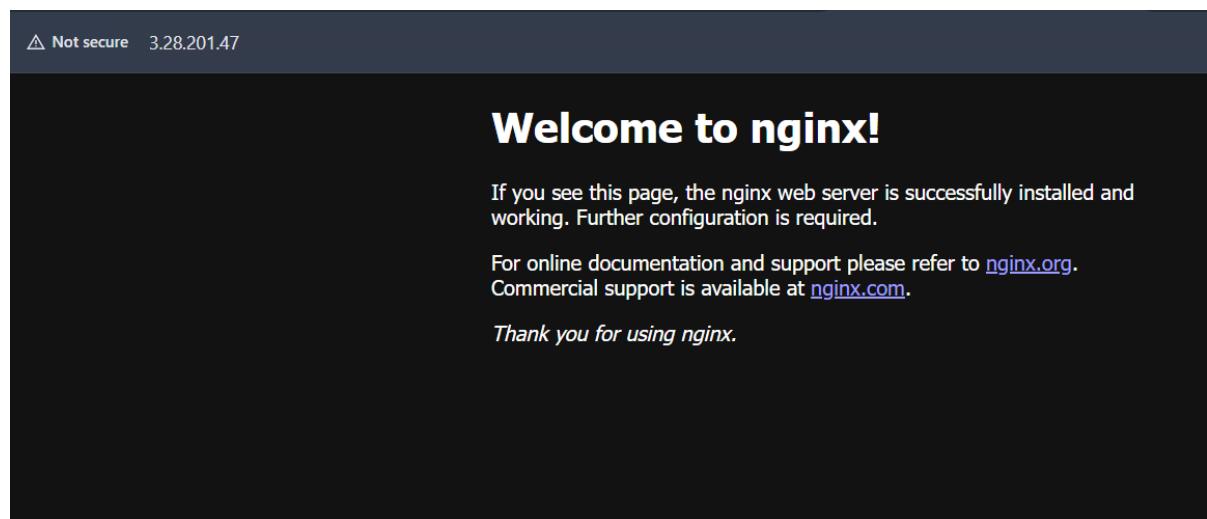
TASK [install nginx and update cache] ****
changed: [3.28.201.47]

TASK [start nginx server] ****
changed: [3.28.201.47]

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

task4_nginx_browser_droplet.png

Nginx Default Page Verified on Droplet Instance



Task 5 – Single Nginx Target Group & HTTPS Prerequisites

task5 project ansible cfg created.png

Project-Level Ansible Configuration File Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ touch ansible.cfg
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano ansible.cfg
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano ansible.cfg
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat ansible.cfg
[defaults]
host_key_checking=False
interpreter_python=/usr/bin/python3
```

task5_main_tf_count_1.png

Terraform Updated to Deploy a Single EC2 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

  # Scale to 3 instances
  count = 1

  instance_suffix = count.index
}
```

task5_terraform_apply_one_instance.png

Terraform Apply Completed – One EC2 Instance Running

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

Outputs:

webserver_public_ips = [
  "3.28.41.237",
]
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task5_terraform_output_single_ip.png

Terraform Output Showing Single EC2 Public IP

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.41.237",
]
```

task5_hosts_nginx_group.png

Ansible Inventory Updated with Nginx Host Group

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano hosts
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat hosts
[nginx]
3.28.41.237

[nginx:vars]
ansible_user=ec2-user
ansible_ssh_private_key_file=~/ssh/id_ed25519
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task5_my_playbook_nginx_group.png

Nginx Playbook Updated with SSL Prerequisites

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible nginx -i hosts -m ping
[WARNING]: Ansible is being run in a world writable directory (/workspaces/terraform_machine.cfg source. For more information see https://docs.ansible.com/ansible-devel/reference_applications/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.41.237 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task5_ansible_play_nginx_group.png

Ansible Playbook Successfully Executed on Nginx Host

```
@SadafRiaz-077 →/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 config file. For more information see https://docs.ansible.com/ansible-devel/reference_appendices/interpreter_discovery.html for more information.
ok: [world-writable-dir]

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.41.237' is using the discovered Python interpreter at '/usr/bin/python3.9', but 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.
ok: [3.28.41.237]

TASK [install nginx and update cache] ****
ok: [3.28.41.237]

TASK [install openssl] ****
ok: [3.28.41.237]

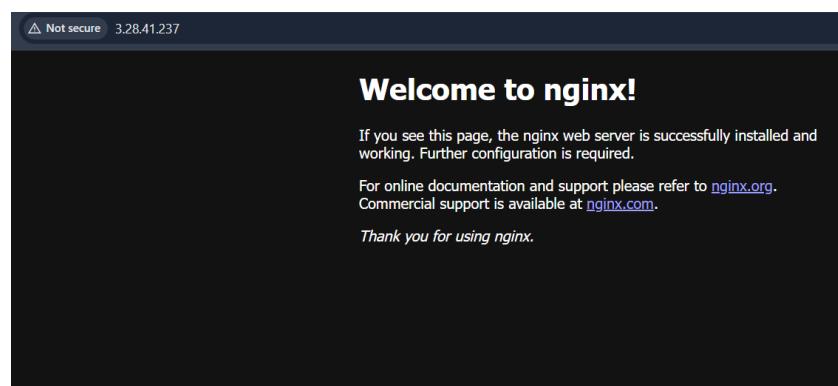
TASK [start nginx server] ****
changed: [3.28.41.237]

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

@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task5_nginx_browser_single.png

Nginx Default Page Verified on Single EC2 Instance



Task 6 – Ansible-Managed SSL Certificates

task6_my_playbook_ssl_section.png

SSL Certificate Generation Section Added to Playbook

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano my-playbook.yaml
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml
- 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: imdsv2_token

    - name: Get current public IP
      uri:
```

task6_ansible_play_ssl.png

Ansible Playbook Executed for SSL Configuration

```
ok: [3.28.41.237]

TASK [Create SSL private directory] *****
changed: [3.28.41.237]

TASK [Create SSL certs directory] *****
changed: [3.28.41.237]

TASK [Get IMDSv2 token] *****
ok: [3.28.41.237]

TASK [Get current public IP] *****
ok: [3.28.41.237]

TASK [Show current public IP] *****
ok: [3.28.41.237] => {
    "msg": "Public IP: 3.28.41.237"
}

TASK [Generate self-signed SSL certificate] *****
changed: [3.28.41.237]

PLAY RECAP *****
3.28.41.237 : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

@SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task6_ssl_cert_file.png

Self-Signed SSL Certificate Verified

```
@sadafRiaz-077 → /workspaces/terraform_machine (main) $ ssh ec2-user@3.28.41.237 -i ~/.ssh/id_ed25519
      _#
  ~\  #####
  ~~ \##### Amazon Linux 2023
  ~~ \#####
  ~~ \###|
  ~~ \#/. https://aws.amazon.com/linux/amazon-linux-2023
  ~~ V~ .-.
  ~~ /_
  ~~ ./
  ~~ /_/
  ~~ /m/
Last login: Tue Jan  6 17:56:14 2026 from 4.240.18.230
[ec2-user@ip-10-0-10-243 ~]$
```

```
Last login: Tue Jan  6 17:56:14 2026 from 4.240.18.230
[ec2-user@ip-10-0-10-243 ~]$ sudo cat /etc/ssl/certs/selfsigned.crt
-----BEGIN CERTIFICATE-----
MIID0zCCAiOgAwIBAgIUV0lh+TpkWsD1292dxgLhr9Iq60QwDQYJKoZIhvcNAQEL
BQAwfjEUMBIGA1UEAwLMy4OC40MSAyMzcwHcNMjYWMTA2MTc1NjE0WhCNMjcw
MTA2MTc1NjE0WhjAWMRQwEgYDVQDDAszLj14ljQxLjIzNzCCASiDQYJKoZIhvcN
AQEBBQADggEPADCCAQoCggEBALFNVS2au4nJDAFBaVRugxHt1jtD2qcR91TJFdLO
oD5V8VcpMTzjxUn1no8qb70YEUZ0zv3RFj2tpbA8LgNYD6rPhv/Z9HaNXSHwVfJ
vKOGC++Baeyyfaah/VdGxqrFd1x5UEU6zTv0HJIy16rdupqlQv+P3E7fJc4+ODz
JakuN5FkpeS2spTiUoU7JCbxeE6R3kCDF5qBoo2dI6f15tZnFEJEsbp+uGx/p
yCRaxTczb6YzC172j/wCcofquEj6XIVeii2zFQRlgGo80w910jjfjAdDbE4s/
0PBilpUi1z2zFHH01PYHi:iWPQTz/b1EqTB1uJq4xtY7mdXMCAwEAAA0BgDB+MB0G
A1UdDgQMBRUoxjFPKVQocCcLNJffjQk2C9FYIjAfBgNVISMEGDAwBRUoxjFPKVQ
oCcLNJffjQk2C9FYIjAPBgNVHREECDAGhWQDHcntMAkGA1UdEwQCMAwCwDVR0P
BAQDAGwIBMGA1UdJQQMAoGCCsGAQUFBwMBMA0GCSqGSIb3DQEBCwUAA4IBAQAS
F+U7z/FI4czP2LsSoWUD3YE0TyAvsDe8j6RaQaP2P1XwTaxF14ZUTWxonscTmjD4u
76spZXh145PhRZSzqMBH5WiSzol8fYXwSEXL/mUx0qo1cP0pLFgmKjRmE4u3qhyB
SfgPovyl/5CcnA4Tl48XB4u1VsDIgcZwNJuyT0Vg48mqj60nI3DNg5HoSFZanvjj
7WN2C81UeyorjWjtIhtOowDlhVXLQ450zrmso6hhG1PLBqBRY+IZe5QnJN6zNcwRs
IKNn/92UYmR9tKyxKVPCzv9v+d5kku16qKGuJyB76nQ7iEdjgjMA/5TiR1CE9kp0
Wp1WyTmZDB/FojG0DxKK
-----END CERTIFICATE-----
[ec2-user@ip-10-0-10-243 ~]$
```

task6_ssl_key_file.png

Private SSL Key Verified on EC2 Instance

```
[ec2-user@ip-10-0-10-243 ~]$ sudo cat /etc/ssl/private/selfsigned.key
-----BEGIN PRIVATE KEY-----
MIIEVQIBADANBgkqhkiG9w0BAQEFAASCBKcwggsjAgEAAoIBAQcxTVUtmruJyQwB
WwFub0Mr7dsb99neFduyRXSzqa+VFFqTE848VJ5fZ5PKm+9BFGfdM79Ry9raW
wPC4DWA+qz4b/2fr2jvoh8FxYbyjhnPvglnssn2mh/1XR16qxpxdcveBFV0s0798y
SMouq3baqqZUL/j9x03yXOpJgbylpMdeRsqRLdrnUlvbvyQgc1hokd5Ag3+aga
DtrnSo4nSLWzxBsrLgz/rhsf6cgkwsU3M2+mHGS09ifweY1ggjn6rh1+lYlxIots
xUEs4BqPNMPZTiy34wHQ2x0L99UfpaVYimdsxr9Nt2B4o1k008/25RKwdhiau
Mb05nVzAgMAECggEAB9Hj2fUT0UwBmguz/S/+xb0whZxjfJfwlyhtLhuNiJM
nuSfdKL8tryeIi1aeuk4tx1axc0t7g4vn2TEGMSpCX09R/dvtURttu5vTHCp
K16xV+e1cANe cpt5ccm+LndxSwJ7c9j1j8jTfgTc/V0nb18adevUgkjIPXMaUSR
MNC1iviPzs7ba87TyceITEM1BKVgZHKpt/2IW1QTSP1k7DEq+Uy3uCL//B8yp
j9sdNzvPvGzIyZ2n+UKTXIH7lcnV0001pHeGZs0e07Jwc1dzkmzUmLAGz/W8mji
Y/uaK1BundkZyySPhncfztAwbty1AhJJhKqNB+LIQKBgQDZ7cmGLhbbuUrqtTsrd
OE7NSA17K2CwIAy6iexQNaC5VcugkIeP9xExs+rVuyns841RHICRNUFN/fAnh
VNufrNgx+xyLo85+ruqbvdnechOKvqqu8gc/L1qC+lr05qfAHsdrqqw7wv+Tbjjs
fq1du3DK1W0KKrFicMV0ThoQKbgDGSdAvunVsb9vGT/h1C00YV+zCFlyPrmz
gJDLhZ7MDsrqfxgDMDGBUAfTuNeFc1l39wsJ4Wq2v00YUuXeMKpL+1m1LY
J3bQlebaBc/jzQd0zMu/3Srav8UGwdsZHMrihCNSVVyK6Rmakm6F1dvOPgyohqTp8
fbMaEhsnkukBqCyzH5Rfdx7ED8f58wf7NH3z2u1wMlogJkywsduheu/Luvub5
Tskme0UflpJNkt8ZTpK0QG98uUF+BLFBQj5GPTwT/YrGLAxI2/0A9BTsckJ/6dr
BH2+y7rsn3hafvnzJpxY8/RDn01YX/GGAmplRn9D5PrzQ0HqJcZc+wBaogBAjqi
M3tOnyA9YlT03MTa8KvhCwm81JGL0apUQScwb52vzfXdsx8ghmmUBGZ1+6knJkh
sr1llpixai0g0Fitq1yaW91DB4oEjXm0n7Fr31y0dgJCrzHspm2tsSP4Pj0evP5
PC4UDLXlmP3fsoosJ79oAkzuoo3ut9z8mKjxAoGAdagY3cjoTnZCMBNIsnZp
321sv6hz5RGz8xJTFw9rkbi18qR824mewQInqc1wrzuyDscwKEi3vxfoeaID8Vxy
mh0oG14sh2IKKMPHQt1tsvhsoi6ub19FEmjGUelWbq8oVuuumYCxzV0H1JKQ1s0
H0YgAvtPZRacsMY8jRuMMBc=
-----END PRIVATE KEY-----
[ec2-user@ip-10-0-10-243 ~]$
```

Task 7 – PHP Front-End Deployment with Templates

task7_files_templates_created.png

Files and Templates Directory Structure Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $      mkdir -p files templates
touch files/index.php
touch templates/nginx.conf.j2
ls -R
.:
README.md    files  locals.tf  modules          outputs.tf  terraform.tfstate      terraform.tfvars
ansible.cfg   hosts   main.tf   my-playbook.yaml  templates  terraform.tfstate.backup  variables.tf

./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
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task7_index_php_content.png

PHP Metadata Web Application Implemented

```
}
```

```
.info a {
  color: white;
  text-decoration: none;
  font-weight: normal;
}
.info a:hover {
  text-decoration: underline;
}
</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>
```

task7_nginx_conf_template.png

Nginx Configuration Template Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano templates/nginx.conf.j2
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat templates/nginx.conf.j2
server_name {{ server_public_ip }};
```

task7_my_playbook_web_deploy.png

Playbook Updated to Deploy PHP Application

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano my-playbook.yaml
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml
- name: Deploy Nginx website and configuration files
  hosts: nginx
  become: true
  vars:
    server_public_ip: "{{ ansible_host }}"
  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
```

task7_ansible_play_web_deploy.png

Ansible Playbook Successfully Deployed PHP Website

```
PLAY [Deploy Nginx website and configuration files] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.41.237' 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.
ok: [3.28.41.237]

TASK [install php-fpm and php-curl] ****
ok: [3.28.41.237]

TASK [Copy website files] ****
ok: [3.28.41.237]

TASK [Copy nginx.conf template] ****
changed: [3.28.41.237]

TASK [Restart nginx] ****
changed: [3.28.41.237]

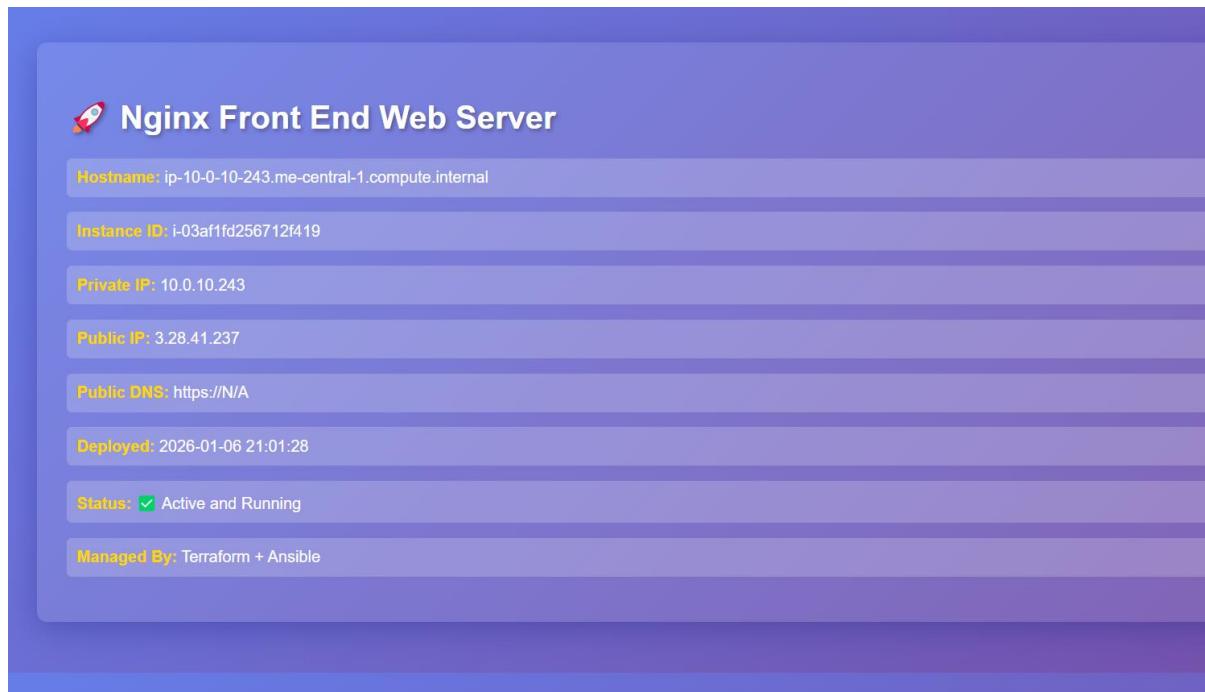
TASK [Start and enable php-fpm] ****
ok: [3.28.41.237]

PLAY RECAP ****
3.28.41.237 : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task7_php_https_browser.png

PHP Application Verified Over HTTPS



Task 8 – Docker & Docker Compose Provisioning via Ansible

task8_terraform_destroy_old.png

Previous Terraform Infrastructure Destroyed

```
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction complete after 0s
module.myapp-webserver[0].aws_security_group.web_sg: Destruction complete after 0s
aws_vpc.myapp_vpc: Destroying... [id=vgw-01b2384c183f51b69]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task8_terraform_apply_docker_instance.png

New EC2 Instance Created for Docker Deployment

```
Apply complete! Resources: 5 added, 0 changed, 0 destroyed.

Outputs:

webserver_public_ips = [
    "158.252.35.123",
]
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task8_terraform_output_new_ip.png

Terraform Output Showing New EC2 Public IP

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform output
webserver_public_ips =
  "158.252.35.123",
]
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task8_hosts_docker_servers.png

Ansible Inventory Updated for Docker Server Group

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ nano hosts.ini
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat hosts.ini
[docker_servers]
158.252.35.123

[docker_servers:vars]
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_user=ec2-user
ansible_python_interpreter=/usr/local/python3.11/bin/python3.11
ansible_ssh_common_args=' -o StrictHostKeyChecking=no'

@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task8_my_playbook_docker.png

Docker & Docker Compose Ansible Playbook Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml
- name: Install Docker Compose
  hosts: docker_servers
  become: 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 system architecture
      debug:
        msg: "System architecture of {{ inventory_hostname }} is {{ ansible_facts['architecture'] }}"

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

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

task8_ansible_play_docker.png

Docker and Docker Compose Installed Successfully

```
TASK [Download Docker Compose] ****
changed: [158.252.35.123]

TASK [Verify Docker Compose installation] ****
changed: [158.252.35.123]

PLAY RECAP ****
158.252.35.123 : ok=9    changed=7    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

@SadafRiaz-077 → /workspaces/terraform_machine (main) $ []
```

task8_docker_ps_remote.png

Docker Verified Using docker ps Command

```
@SadafRiaz-077 → /workspaces/terraform_machine (main) $ ssh -i ~/.ssh/id_ed25519 ec2-user@158.252.35.123
Last login: Sat Jan 10 10:31:51 2026 from 4.240.39.199
,
~\_ #_
~~ \_\#\#\#_      Amazon Linux 2
~~ \_\#\#\#\\
~~ \#\#\|      AL2 End of Life is 2026-06-30.
~~ \#/      --
~~ V~' '-->
~~ /      A newer version of Amazon Linux is available!
~~ ._. /      Amazon Linux 2023, GA and supported until 2028-03-15.
~~ /_ /      https://aws.amazon.com/linux/amazon-linux-2023/
~/m/         

No packages needed for security; 6 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-20-175 ~]$ sudo systemctl start docker
[ec2-user@ip-10-0-20-175 ~]$ sudo systemctl enable docker
[ec2-user@ip-10-0-20-175 ~]$ docker --version
Docker version 25.0.14, build 0bab007
[ec2-user@ip-10-0-20-175 ~]$ docker compose version
Docker Compose version v5.0.1
[ec2-user@ip-10-0-20-175 ~]$ []
```

Task 9 – Gitea Docker Stack via Ansible & Terraform

task9_my_playbook_add_user_to_docker.png

```
~ webserver_public_ips = [
  + (known after apply),
]
module.myapp-webserver.aws_instance.webserver: Creating...
module.myapp-webserver.aws_instance.webserver: Still creating... [10s elapsed]
module.myapp-webserver.aws_instance.webserver: Creation complete after 13s [id=i-06f2ca80738f568f2]

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

Outputs:

webserver_public_ips = [
  "3.29.125.248",
]
@SadafRiaz-077 → /workspaces/terraform_machine (main) $ []
```

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml
    state: restarted

- 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
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task9_project_vars.png

Project Variables File Created

```
▶ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat project-vars.yaml
normal_user: ec2-user
docker_compose_file_location: /workspaces/terraform_machine
▶ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat compose.yaml
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
expose:
  - 5432
networks:
  - webnet

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

networks:
  webnet:
    name: webnet
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task9_my_playbook_deploy_containers.png

Playbook Updated to Deploy Docker Containers

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat my-playbook.yaml
- name: fail if docker is not accessible
  fail:
    msg: "Docker is not accessible on this host"
    when: docker_ps.rc != 0

# 2Deploy Docker Containers
- 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.stderr"
      ignore_errors: true

@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task9_ansible_play_gitea.png

Gitea and Database Containers Deployed Successfully

```
changed: [158.252.35.123]

TASK [Deploy containers using docker-compose] *****
ok: [158.252.35.123]

TASK [Show docker-compose output] *****
ok: [158.252.35.123] => {
    "compose_result.stdout": ""
}

PLAY RECAP *****
158.252.35.123 : ok=9    changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task9_sg_ingress_3000.png

Security Group Updated to Allow Port 3000

```

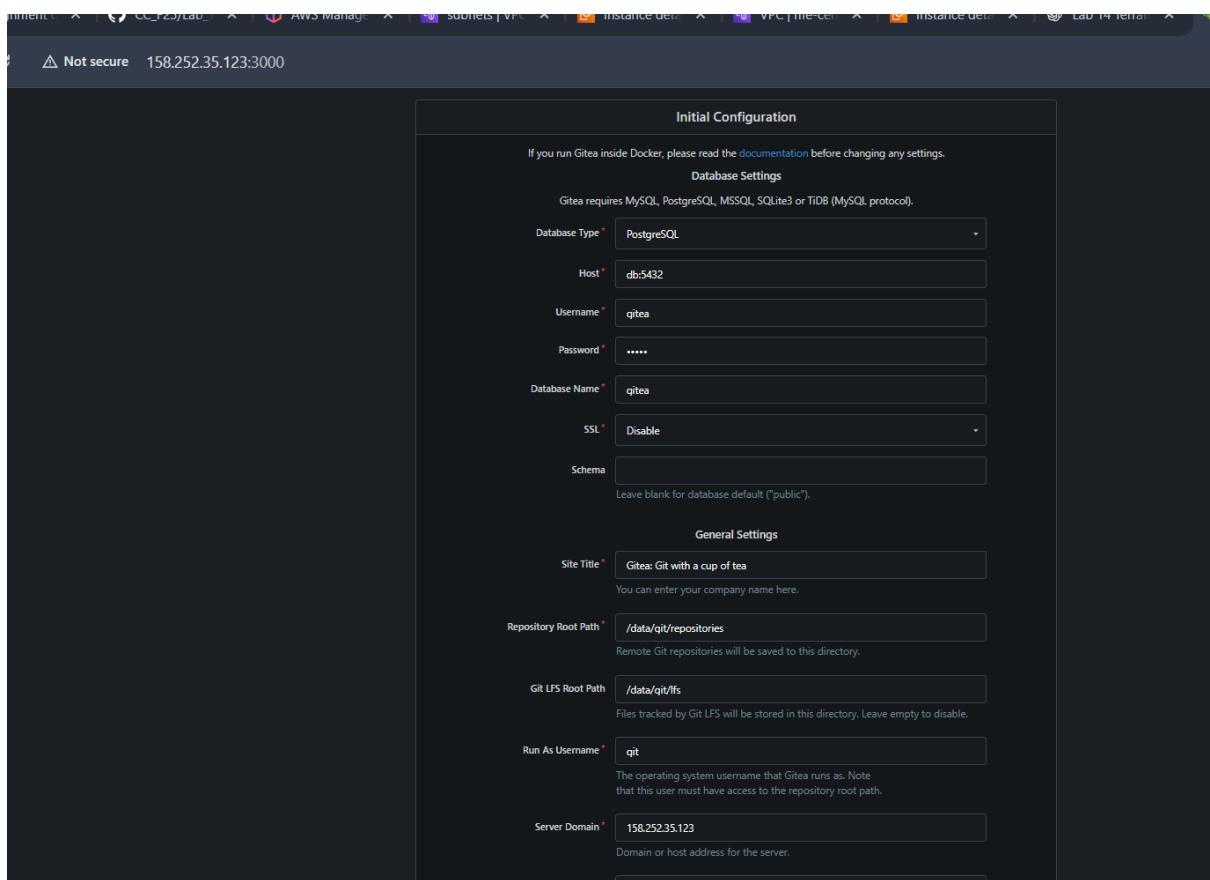
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/

No packages needed for security; 6 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-20-175 ~]$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
f8589f130189 gitea/gitea:latest "/usr/bin/entrypoint..." About a minute ago Up About a minute 22/tcp, 0.0.0.0:3000->3000/tcp, ::1:3000->3000/tcp
5eae232051d7 postgres:alpine "docker-entrypoint.s..." About a minute ago Up About a minute 5432/tcp
[ec2-user@ip-10-0-20-175 ~]$ []

```

task9_gitea_browser.png

Gitea Web Interface Verified in Browser



Task 10 – Automating Ansible with Terraform (null_resource)

task10_null_resource_main_tf.png

Terraform null_resource Added to Trigger Ansible

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat main.tf
}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
  }
}
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task10_terraform_destroy_before_null.png

Infrastructure Destroyed Before Testing Automation

```
module.myapp-webserver[0].aws_key_pair.ssh-key: Destruction
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destruction
module.myapp-webserver[0].aws_security_group.web_sg: Destruction
aws_vpc.myapp_vpc: Destroying... [id=vpc-09123e1c621bae27d]
aws_vpc.myapp_vpc: Destruction complete after 0s

Destroy complete! Resources: 7 destroyed.
@SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

task10_terraform_apply_after_wait.png

Terraform Apply Successful After SSH Wait Fix

```
Plan: 4 to add, 0 to change, 0 to destroy.
aws_vpc.myapp_vpc: Creating...
aws_vpc.myapp_vpc: Creation complete after 1s [id=vpc-0a6c66db87c322a20]
aws_subnet.myapp_subnet: Creating...
aws_security_group.web_sg: Creating...
aws_subnet.myapp_subnet: Creation complete after 1s [id=subnet-0407ef2c494be4647]
aws_security_group.web_sg: Creation complete after 3s [id=sg-0a63f1ab00d72f88a]
aws_instance.webserver: Creating...
aws_instance.webserver: Still creating... [10s elapsed]
aws_instance.webserver: Creation complete after 12s [id=i-00d14c18a8fef67d4]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.
```

task10_app_browser_post_null_resource.png

Application Verified After Terraform + Ansible Automation

Initial Configuration

If you run Gitea inside Docker, please read the [documentation](#) before changing any settings.

Database Settings

Gitea requires MySQL, PostgreSQL, MSSQL, SQLite3 or TiDB (MySQL protocol).

Database Type:	PostgreSQL
Host:	db:5432
Username:	gitea
Password:
Database Name:	gitea
SSL:	Disable
Schema:	Leave blank for database default ("public").

General Settings

Site Title:	Gitea: Git with a cup of tea
You can enter your company name here.	
Repository Root Path:	/data/gitea/repositories
Remote Git repositories will be saved to this directory.	
Git LFS Root Path:	/data/gitea/lfs
Files tracked by Git LFS will be stored in this directory. Leave empty to disable.	
Run As Username:	git
The operating system username that Gitea runs as. Note: this user must have access to the repository root path.	

Task 11 – Dynamic Inventory with aws_ec2 Plugin

task11_ansible_cfg_aws_ec2.png

Ansible Configured to Use aws_ec2 Inventory Plugin

```
@SadafRiaz-077 →/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=/home/codespace/.ssh/id_ed25519
```

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task11_inventory_aws_ec2_created.png

AWS EC2 Dynamic Inventory File Created

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ touch inventory_aws_ec2.yaml
ls -la inventory_aws_ec2.yaml
-rw-rw-rw- 1 codespace codespace 0 Jan 10 13:59 inventory_aws_ec2.yaml
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ █
```

task11_inventory_aws_ec2_initial.png

Initial aws_ec2 Inventory Configuration

```
└--  
  plugin: aws_ec2  
  regions:  
    - me-central-1  
  boto_profile: default  
  filters:  
    instance-state-name: running  
  keyed_groups:  
    - key: tags.Env  
      prefix: env  
    - key: tags.Role  
      prefix: role  
  cache: yes  
  cache_plugin: jsonfile  
  cache_timeout: 300  
  cache_dir: ./inventory_cache
```

task11_main_tf_dev_prod_modules.png

Terraform Configured for Dev and Prod EC2 Instances

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat main.tf  
data "http" "my_ip" {  
  url = "https://ifconfig.me"  
}  
  
locals {  
  my_ip = trimspace(data.http.my_ip.body)  
}  
  
# -----  
# Optional: Ansible automation via null_resource  
# -----  
resource "null_resource" "configure_server" {  
  triggers = {
```

task11_outputs_tf_dev_prod_ips.png

Terraform Outputs Added for Dev and Prod Instances

```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat outputs.tf  
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]  
}  
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ []
```

task11_terraform_apply_dynamic_setup.png

Terraform Applied for Dynamic Inventory Setup

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ terraform apply -auto-approve
]
aws_vpc.myapp_vpc: Creating...
aws_vpc.myapp_vpc: Still creating... [10s elapsed]
aws_vpc.myapp_vpc: Creation complete after 12s [id=vpc-0ef1d132cbeada805]
module.myapp-subnet.aws_subnet.subnet: Creating...
module.myapp-subnet.aws_subnet.subnet: Still creating... [10s elapsed]
module.myapp-subnet.aws_subnet.subnet: Creation complete after 11s [id=subnet-08e1050a80e6c434b]
module.myapp-webserver-prod[0].aws_instance.webserver[0]: Creating...
module.myapp-webserver[0].aws_instance.webserver[0]: Creating...
module.myapp-webserver[0].aws_instance.webserver[0]: Still creating... [10s elapsed]
module.myapp-webserver[0].aws_instance.webserver[0]: Still creating... [10s elapsed]
module.myapp-webserver-prod[0].aws_instance.webserver[0]: Creation complete after 13s [id=i-0bc8babe48292f20f]
module.myapp-webserver[0].aws_instance.webserver[0]: Creation complete after 13s [id=i-0966ed5b382c20ae8]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.

Outputs:

prod_webserver_public_ips = [
  [
    "3.28.132.64",
  ],
]
webserver_public_ips = [
  [
    "51.112.253.140",
  ],
]
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ]
```

task11_terraform_output_dynamic_ips.png

Terraform Output Showing Dev and Prod Public Ips

- @SadafRiaz-077 →/workspaces/terraform_machine (main) \$ terraform output
prod_webserver_public_ips = [
 [
 "3.28.132.64",
],
]
webserver_public_ips = [
 [
 "51.112.253.140",
],
]

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ]
```

task11_boto_install.png

boto3 and botocore Installed Successfully

task11_boto_version.png

boto3 Version Verified

```
Successfully installed boto3-1.42.25 botocore-1.42.25 jmespath-1.0.1 s3transfer-0.16.0
@sadafriaz-077 → /workspaces/terraform_machine (main) $ $(which python3) -c "import boto3, botocore; print(boto3.__version__)"
1.42.25
@sadafriaz-077 → /workspaces/terraform_machine (main) $ 
```

task11 ansible inventory graph initial.png

Dynamic Inventory Graph Generated Successfully

```
● @SadaFRiaz-077 → /workspaces/terraform_machine (main) $ ansible-inventory -i inventory_aws_ec2.yaml --graph  
[WARNING]: Collection amazon.aws does not support Ansible version 2.15.12  
@all:  
  |--@ungrouped:  
  |--@aws_ec2:  
    |  |--ip-10-0-1-110.me-central-1.compute.internal  
    |  |--ip-10-0-20-29.me-central-1.compute.internal  
    |  |--ip-10-0-1-160.me-central-1.compute.internal  
    |  |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com  
    |  |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
```

Task 12 – Filtering EC2 Instances by Tags & Instance Type

task12 inventory aws ec2 tag groups.png

Dynamic Inventory Grouping Enabled Using EC2 Tags

```
● @SadafRiaz-077 → /workspaces/terraform_machine (main) $ cat inventory_aws_ec2.yaml
---
plugin: aws_ec2
regions:
  - me-central-1

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

○ @SadafRiaz-077 → /workspaces/terraform_machine (main) $ █
```

task12_inventory_graph_tag_groups.png

Inventory Graph Showing Tag-Based Groups

```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible-inventory --graph
[WARNING]: Collection amazon.aws does not support Ansible version 2.15.12
@all:
  |--@ungrouped:
  |--@aws_ec2:
    |--ip-10-0-1-110.me-central-1.compute.internal
    |--ip-10-0-20-29.me-central-1.compute.internal
    |--ip-10-0-1-160.me-central-1.compute.internal
    |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com
    |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
  |--@tag_Name_dev_ec2_0:
    |--ip-10-0-1-110.me-central-1.compute.internal
  |--@tag_Name_dev_webserver:
    |--ip-10-0-20-29.me-central-1.compute.internal
  |--@tag_Name_dev_ec2_1:
    |--ip-10-0-1-160.me-central-1.compute.internal
  |--@tag_Name_dev_webserver_0:
    |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com
  |--@tag_Name_prod_webserver_0:
    |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task12_inventory_aws_ec2_instance_type_groups.png

Dynamic Inventory Extended to Group by Instance Type

```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat inventory_aws_ec2.yaml
---
plugin: aws_ec2
regions:
  - me-central-1

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

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

```
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $ 
```

task12_inventory_graph_full.png

Inventory Graph Showing Tag and Instance Type Groups

```

@SadafRiaz-077 →/workspaces/terraform_machine (main) $ ansible-inventory --graph
[WARNING]: Collection amazon.aws does not support Ansible version 2.15.12
@all:
  |--@ungrouped:
  |--@aws_ec2:
    |--ip-10-0-1-110.me-central-1.compute.internal
    |--ip-10-0-20-29.me-central-1.compute.internal
    |--ip-10-0-1-160.me-central-1.compute.internal
    |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com
    |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
  |--@tag_Name_dev_ec2_0:
    |--ip-10-0-1-110.me-central-1.compute.internal
  |--@instance_type_t3_micro:
    |--ip-10-0-1-110.me-central-1.compute.internal
    |--ip-10-0-20-29.me-central-1.compute.internal
    |--ip-10-0-1-160.me-central-1.compute.internal
    |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com
  |--@tag_Name_dev_webserver:
    |--ip-10-0-20-29.me-central-1.compute.internal
  |--@tag_Name_dev_ec2_1:
    |--ip-10-0-1-160.me-central-1.compute.internal
  |--@tag_Name_dev_webserver_0:
    |--ec2-51-112-253-140.me-central-1.compute.amazonaws.com
  |--@tag_Name_prod_webserver_0:
    |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
  |--@instance_type_t3_nano:
    |--ec2-3-28-132-64.me-central-1.compute.amazonaws.com
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ []

```

task12_my_playbook_all_hosts.png

Playbook Configured to Run on All Hosts

```

TASK [Install openssl] ****
ok: [ec2-40-172-191-130]

TASK [Start and enable nginx] ****
ok: [ec2-40-172-191-130]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-191-130]

PLAY [Configure SSL certificates] ****

TASK [Gathering Facts] ****
ok: [ec2-40-172-191-130]

TASK [Create SSL directories] ****
ok: [ec2-40-172-191-130] => (item={'path': '/etc/ssl/private', 'mode': '0700'})
ok: [ec2-40-172-191-130] => (item={'path': '/etc/ssl/certs', 'mode': '0755'})

TASK [Generate self-signed SSL certificate (Amazon Linux compatible)] ****
changed: [ec2-40-172-191-130]

PLAY RECAP ****
ec2-40-172-191-130 : ok=10   changed=3   unreachable=0   failed=0   skipped=0   rescued=0   
```

task12_ansible_play_all.png

Playbook Executed on All Dev and Prod Instances

```
TASK [Generate self-signed SSL certificate (Amazon Linux compatible)] ****
changed: [ec2-40-172-191-130]

PLAY RECAP ****
ec2-40-172-191-130 : ok=10    changed=3    unreachable=0    failed=0    skipped=0    rescued=0

@SadafRiaz-077 → /workspaces/terraform_machine (main) $
```

Task 13 – Ansible Roles: nginx, ssl, webapp

task13_main_tf_single_dev.png

Terraform Configured for Single Dev Environment

```
@SadafRiaz-077 → /workspaces/terraform_machine/ansible (main) $ cat main.tf
#####
module "myapp-subnet" {
  source      = "./modules/subnet"
  vpc_id      = aws_vpc.myapp_vpc.id
  subnet_cidr_block = var.subnet_cidr_block
  availability_zone = var.availability_zone
  env_prefix   = var.env_prefix
  default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id
}

#####
# Webserver module (DEV = 1 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

  # Single instance for dev
  count        = 1
  instance_suffix = count.index
}
```

task13_ansible_structure_created.png

Ansible Directory Structure Created

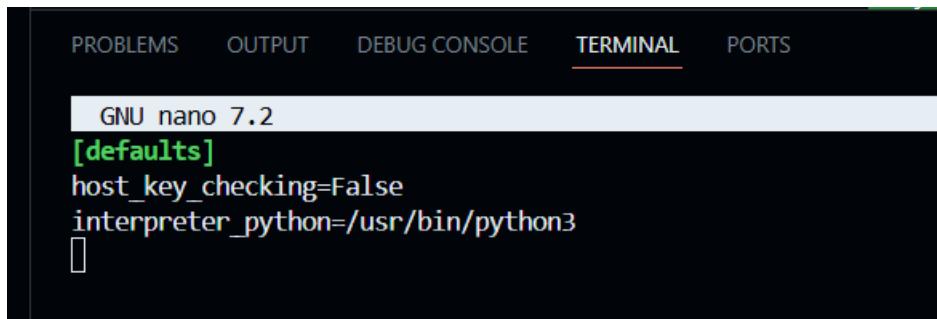
```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ mkdir -p ansible
cd ansible
mkdir inventory roles
touch ansible.cfg my-playbook.yaml
ls -R
.:
ansible.cfg inventory my-playbook.yaml roles

./inventory:

./roles:
○ @SadafRiaz-077 →/workspaces/terraform_machine/ansible (main) $ 
```

task13_ansible_cfg_project.png

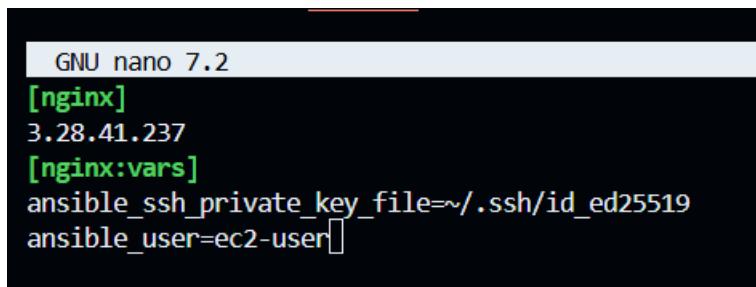
Project-Level Ansible Configuration File



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

task13_ansible_inventory_hosts.png

Ansible Inventory for Nginx Role Created



```
GNU nano 7.2
[nginx]
3.28.41.237
[nginx:vars]
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_user=ec2-user
```

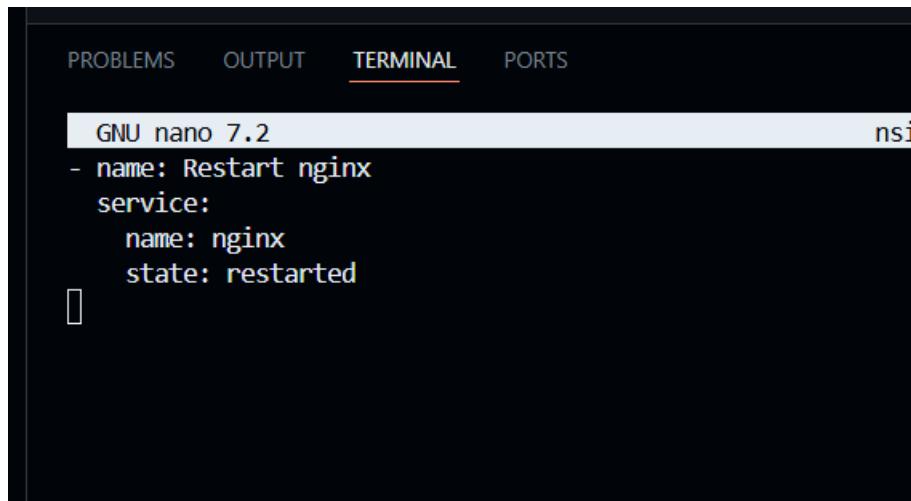
task13_roles_created.png

Nginx, SSL, and WebApp Roles Initialized

```
@SadafRiaz-077 →/workspaces/terraform_machine (main) $ cd ansible/roles
ansible-galaxy role init nginx
ansible-galaxy role init ssl
ansible-galaxy role init webapp
ls -R
```

task13_nginx_handlers_main.png

Nginx Role Handlers Implemented

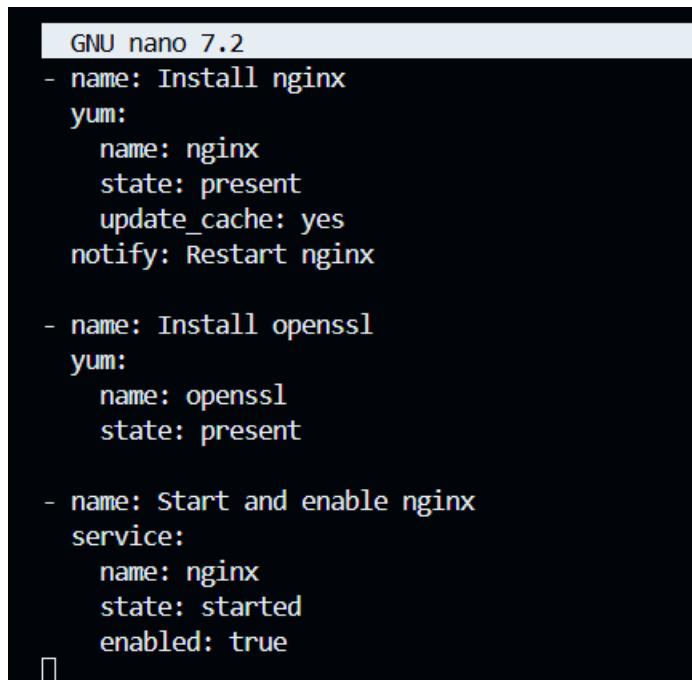


A screenshot of a terminal window titled "TERMINAL". The title bar also includes "PROBLEMS", "OUTPUT", and "PORTS". The terminal content shows the following code:

```
GNU nano 7.2
- name: Restart nginx
  service:
    name: nginx
    state: restarted
```

task13_nginx_tasks_main.png

Nginx Role Tasks Implemented



A screenshot of a terminal window showing implemented tasks. The terminal content shows the following code:

```
GNU nano 7.2
- 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
```

task13_my_playbook_nginx_only.png

Playbook Using Only Nginx Role

The screenshot shows a terminal window with tabs: PROBLEMS, OUTPUT, TERMINAL (which is underlined), and PORTS. The terminal content is as follows:

```
GNU nano 7.2
---
- name: Deploy NGINX only
  hosts: nginx
  become: true
  roles:
    - nginx
```

task13_ansible_play_nginx_only.png

Nginx Role Executed Successfully

```
ok: [3.28.41.237]
TASK [Create SSL private directory] *****
changed: [3.28.41.237]

TASK [Create SSL certs directory] *****
changed: [3.28.41.237]

TASK [Get IMDSv2 token] *****
ok: [3.28.41.237]

TASK [Get current public IP] *****
ok: [3.28.41.237]

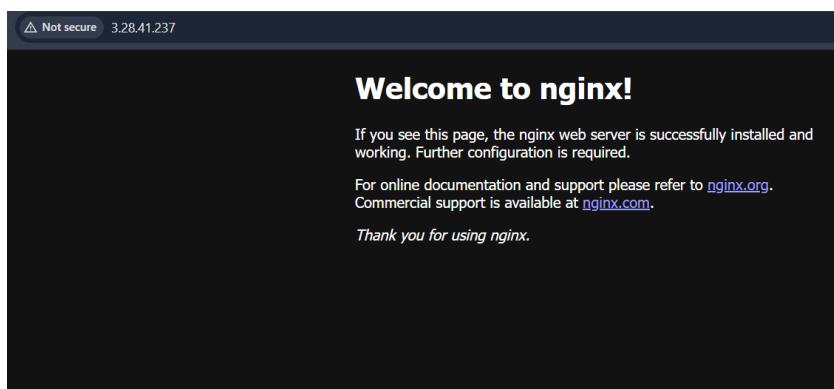
TASK [Show current public IP] *****
ok: [3.28.41.237] => {
  "msg": "Public IP: 3.28.41.237"
}

TASK [Generate self-signed SSL certificate] *****
changed: [3.28.41.237]

PLAY RECAP *****
3.28.41.237 : ok=7    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
@sadafriaz-077 → /workspaces/terraform_machine (main) $
```

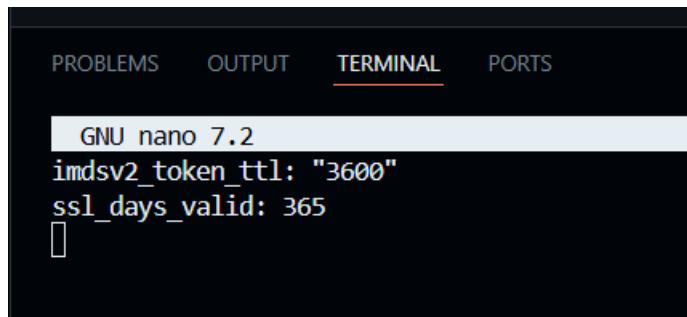
task13_nginx_browser_roles.png

Nginx Verified After Role-Based Deployment



task13_ssl_defaults_main.png

SSL Role Default Variables Defined

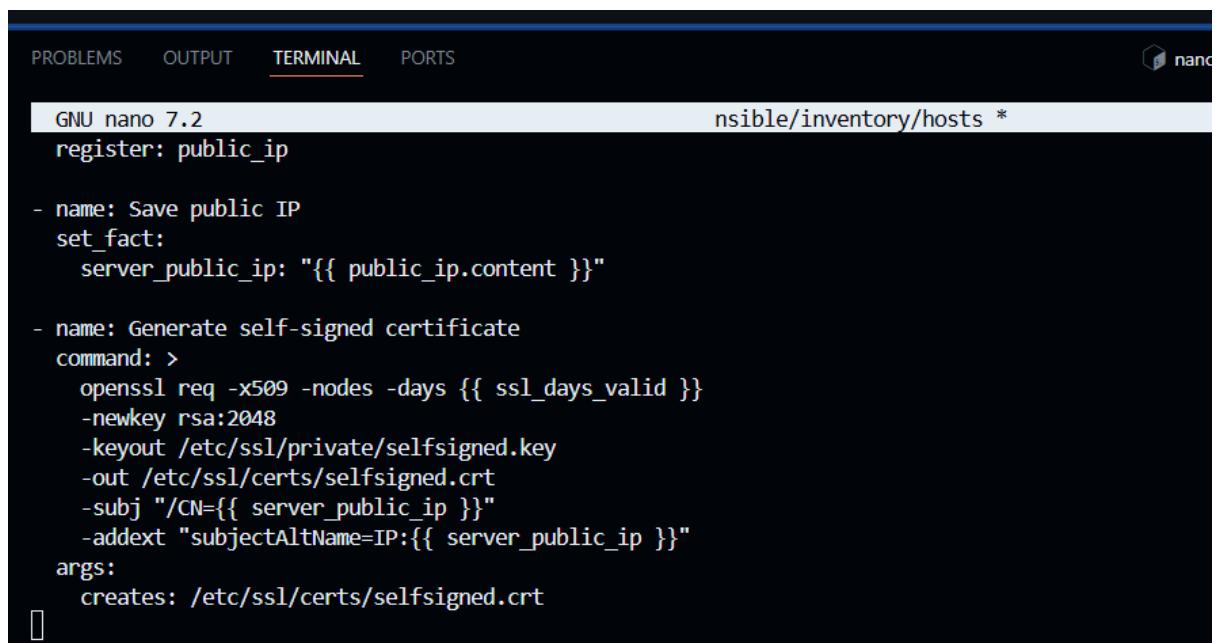


PROBLEMS OUTPUT TERMINAL PORTS

```
GNU nano 7.2
imdsv2_token_ttl: "3600"
ssl_days_valid: 365
```

task13_ssl_tasks_main.png

SSL Role Tasks Implemented



PROBLEMS OUTPUT TERMINAL PORTS

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

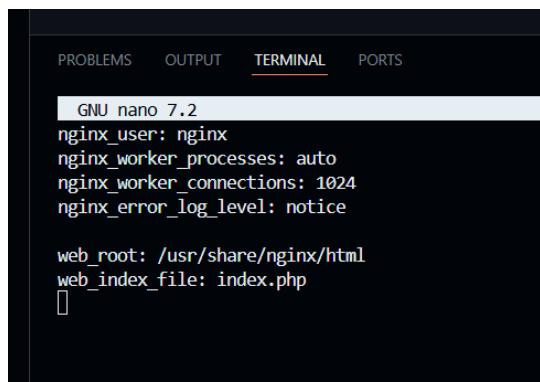
```
register: public_ip

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

- name: Generate self-signed certificate
  command: >
    openssl req -x509 -nodes -days {{ ssl_days_valid }}
    -newkey rsa:2048
    -keyout /etc/ssl/private/selfsigned.key
    -out /etc/ssl/certs/selfsigned.crt
    -subj "/CN={{ server_public_ip }}"
    -addext "subjectAltName=IP:{{ server_public_ip }}"
  args:
    creates: /etc/ssl/certs/selfsigned.crt
```

task13_webapp_defaults_main.png

WebApp Role Default Variables Defined



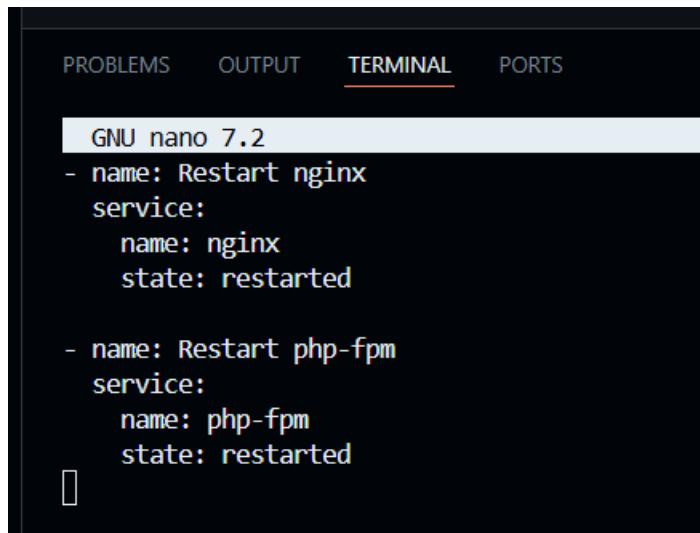
PROBLEMS OUTPUT TERMINAL PORTS

```
GNU nano 7.2
nginx_user: nginx
nginx_worker_processes: auto
nginx_worker_connections: 1024
nginx_error_log_level: notice

web_root: /usr/share/nginx/html
web_index_file: index.php
```

task13_webapp_handlers_main.png

WebApp Role Handlers Implemented

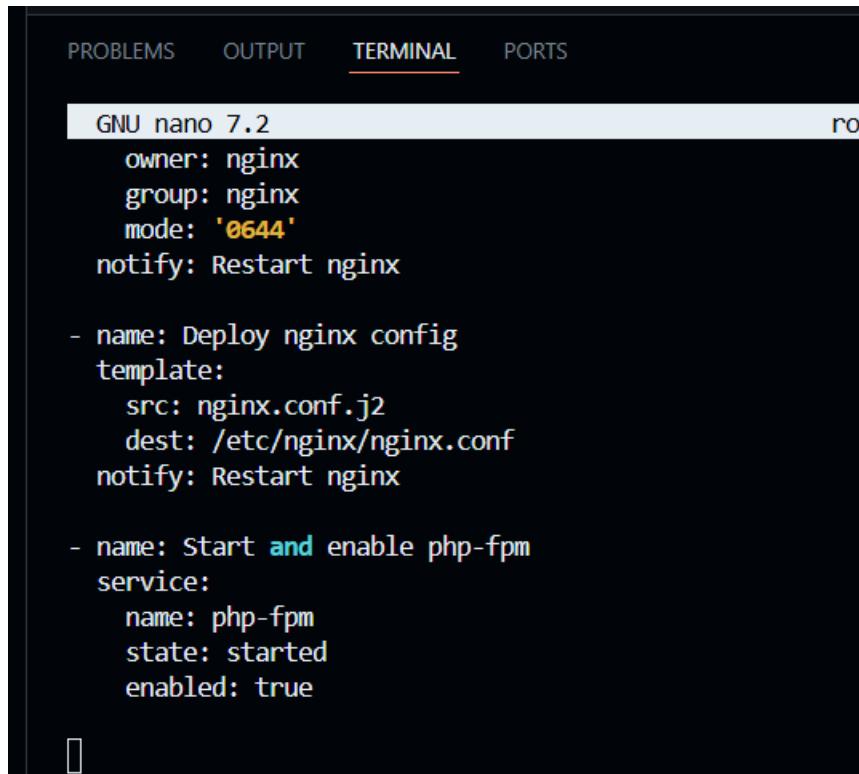


```
GNU nano 7.2
- name: Restart nginx
  service:
    name: nginx
    state: restarted

- name: Restart php-fpm
  service:
    name: php-fpm
    state: restarted
[]
```

task13_webapp_tasks_main.png

WebApp Role Tasks Implemented



```
GNU nano 7.2
  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
[]
```

task13_my_playbook_roles.png

Final Playbook Using Nginx, SSL, and WebApp Roles

```
GNU nano 7.2
---
- name: Deploy NGINX Web Stack with SSL and PHP
  hosts: nginx
  become: true
  roles:
    - nginx
    - ssl
    - webapp
```

task13_ansible_play_roles.png

All Roles Executed Successfully

```
PLAY [Deploy Nginx website and configuration files] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.41.237' 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/2.10/reference_appendices/interpreter_discovery.html for more information.
ok: [3.28.41.237]

TASK [install php-fpm and php-curl] ****
ok: [3.28.41.237]

TASK [Copy website files] ****
ok: [3.28.41.237]

TASK [Copy nginx.conf template] ****
changed: [3.28.41.237]

TASK [Restart nginx] ****
changed: [3.28.41.237]

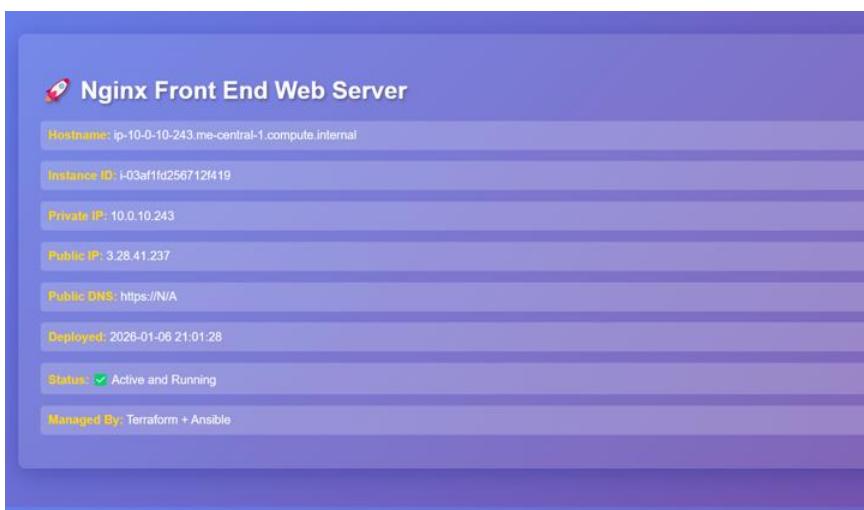
TASK [Start and enable php-fpm] ****
ok: [3.28.41.237]

PLAY RECAP ****
3.28.41.237 : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

@SadafRiaz-077 ~/workspaces/terraform_machine (main) $ 
```

task13_php_https_browser_roles.png

Final PHP Application Verified Over HTTPS



Cleanup

cleanup_terraform_destroy.png

Terraform Infrastructure Destroyed Successfully

```
● @SadafRiaz-077 →/workspaces/terraform_machine (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 complete! Resources: 0 destroyed.
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

cleanup_tfstate.png

Terraform State File Verified Empty

```
● @SadafRiaz-077 →/workspaces/terraform_machine (main) $ cat terraform.tfstate
{
  "version": 4,
  "terraform_version": "1.6.4",
  "serial": 1,
  "lineage": "9e669687-ae02-a336-7f0b-c4aa30a62dbe",
  "outputs": {},
  "resources": [],
  "check_results": null
}
○ @SadafRiaz-077 →/workspaces/terraform_machine (main) $
```

cleanup_aws_console.png

AWS Console Showing No Running EC2 Instances

