



Fatima Jinnah Women University

Opening Portals of Excellence Through Higher Education

CLOUD COMPUTING LAB

BSE (V-B)

LAB 14

Submitted By: Urwa Zahra

Roll No: 2023-BSE-068

Submitted To: Sir Muhammad Shoaib

Task 0 – Lab Setup (Codespace & GH CLI)

The screenshot shows the VS Code interface with the 'TERMINAL' tab selected. In the terminal, the user is in a workspace named 'terraformation_machine'. The file tree on the left shows files like .gitignore, locals.tf, main.tf, outputs.tf, README.md, and variables.tf.

```
@Urwa012 → /workspaces/terraform_machine (main) $
```

Task 1 – Generate ssh key and Initial Terraform apply

```
● @Urwa012 → /workspaces/terraform_machine (main) $ aws --version
aws-cli/2.32.32 Python/3.13.11 Linux/6.8.0-1030-azure exe/x86_64.ubuntu.24
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform --version
Terraform v1.14.3
on linux_amd64
● @Urwa012 → /workspaces/terraform_machine (main) $ ansible --version
ansible [core 2.19.5]
  config file = /etc/ansible/ansible.cfg
  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
  pyyaml version = 6.0.3 (with libyaml v0.2.5)
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ ls ~/.ssh
ls: cannot access '/home/codespace/.ssh': No such file or directory
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ aws configure
AWS Access Key ID [None]: AKIA35PWBHQSN6LJHVHK
AWS Secret Access Key [None]: 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
Default region name [None]: me-central-1
Default output format [None]: json
● @Urwa012 → /workspaces/terraform_machine (main) $ aws sts get-caller-identity
{
    "UserId": "AIDA35PWBHQSN6LJHVHK",
    "Account": "819244121124",
    "Arn": "arn:aws:iam::819244121124:user/Admin"
}
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -N ""
Generating public/private ed25519 key pair.
/home/codespace/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
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:Cr+cRg993BFsGYKxdaAoWG126TSkRD8pV2oTGT9O5k8 codespace@codespaces-65d075
The key's randomart image is:
+--[ED25519 256]--+
| .o++B*+= |
| .. o=**O= |
| o .o=+Oo=. |
| . +.B... |
| . .SoE |
| oo..o. |
| .oo .. |
| ..o. |
| .+ |
+----[SHA256]-----
```

```
+----[SHA256]-----  
● @Urwa012 → /workspaces/terraform_machine (main) $ ls -la ~/.ssh  
total 20  
drwxr-xr-x 2 codespace codespace 4096 Jan  9 22:19 .  
drwxr-x--- 1 codespace codespace 4096 Jan  9 22:19 ..  
-rw----- 1 codespace codespace  419 Jan 10 17:37 id_ed25519  
-rw-r--r-- 1 codespace codespace 109 Jan 10 17:37 id_ed25519.pub
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ touch terraform.tfvars  
● @Urwa012 → /workspaces/terraform_machine (main) $ ls -la terraform.tfvars  
-rw-rw-rw- 1 codespace codespace 219 Jan 10 17:41 terraform.tfvars
```

```
● @Urwa012 → /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"
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform init  
Initializing the backend...  
Initializing modules...  
Initializing provider plugins...  
- Reusing previous version of hashicorp/http from the dependency lock file  
- Reusing previous version of hashicorp/aws from the dependency lock file  
- Using previously-installed hashicorp/http v3.5.0  
- Using previously-installed hashicorp/aws v6.28.0  
  
Terraform has been successfully initialized!  
  
You may now begin working with Terraform. Try running "terraform plan" to see  
any changes that are required for your infrastructure. All Terraform commands  
should now work.  
  
If you ever set or change modules or backend configuration for Terraform,  
rerun this command to reinitialize your working directory. If you forget, other  
commands will detect it and remind you to do so if necessary.
```

```
module.myapp-webserver[0].aws_key_pair.ssh-key: Creation complete after 1s [id=d  
ev-serverkey-0]  
module.myapp-webserver[1].aws_security_group.web_sg: Modifications complete afte  
r 1s [id=sg-058b13df1cea9fa55]  
module.myapp-webserver[0].aws_security_group.web_sg: Modifications complete afte  
r 1s [id=sg-0ealc93c9729b8a0d]
```

```
Apply complete! Resources: 2 added, 2 changed, 2 destroyed.
```

Outputs:

```
webserver_public_ips = [  
    "3.28.46.52",  
    "3.28.41.146",  
]
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform output  
webserver_public_ips = [  
    "3.28.46.52",  
    "3.28.41.146",  
]
```

Task 2 – Static Ansible inventory with two EC2 instances

```
@Urwa012 → /workspaces/terraform_machine (main) $ pipx install ansible-core  
/usr/bin/ansible-playbook  
⚠ Note: ansible-pull was already on your PATH at /usr/bin/ansible-pull  
⚠ Note: ansible-test was already on your PATH at /usr/bin/ansible-test  
⚠ Note: ansible-vault was already on your PATH at /usr/bin/ansible-vault  
installed package ansible-core 2.20.1, installed using Python 3.12.1  
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! 🎉 🎉 🎉  
● @Urwa012 → /workspaces/terraform_machine (main) $ ansible --version  
ansible [core 2.20.1]  
config file = /etc/ansible/ansible.cfg  
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)
```

```
pyyaml version = 6.0.3 (with libyaml v0.2.5)  
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform output  
webserver_public_ips = [  
    "3.28.46.52",  
    "3.28.41.146",  
]
```

```

● @Urwa012 → /workspaces/terraform_machine (main) $ touch hosts
● @Urwa012 → /workspaces/terraform_machine (main) $ ls -la hosts
-rw-rw-rw- 1 codespace codespace 0 Jan 10 17:56 hosts

● @Urwa012 → /workspaces/terraform_machine (main) $ cat > hosts <<EOF
3.28.46.52 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519
3.28.41.146 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519
EOF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat hosts
3.28.46.52 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519
3.28.41.146 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519

```



```

● @Urwa012 → /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}

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

```



```

● @Urwa012 → /workspaces/terraform_machine (main) $ cat > hosts <<EOF
3.28.46.52 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
3.28.41.146 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
EOF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat hosts
3.28.46.52 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'
3.28.41.146 ansible_user=ec2-user ansible_ssh_private_key_file=~/ssh/id_ed25519 ansible_ssh_common_args=' -o StrictHostKeyChecking=no'

```



```

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '3.28.132.53' 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.
3.28.132.53 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '51.112.229.205' 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.
51.112.229.205 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

```

Task 3 - Scale to three instances & group-based inventory

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL FOCUS

GNU nano 7.2 main.tf *

```
source = "./modules/subnet"
vpc_id = aws_vpc.myapp_vpc.id
subnet_cidr_block = var.subnet_cidr_block
availability_zone = var.availability_zone
env_prefix = var.env_prefix
default_route_table_id = aws_vpc.myapp_vpc.default_route_table_id

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

  # Loop count
  count           = 3
  # Use count.index to differentiate instances
  instance_suffix = count.index
}
```

```
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 0s [id=dev-serverkey-2]
module.myapp-webserver[2].aws_security_group.web_sg: Creation complete after 2s [id=sg-035051cc8b8f16fb9]
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 12s [id=i-070a82ff839ceec81]
```

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

Outputs:

```
webserver_public_ips = [
  "3.28.132.53",
  "51.112.229.205",
  "51.112.253.19",
]
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.132.53",
  "51.112.229.205",
  "51.112.253.19",
]
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ cat hosts
[ec2]
3.28.132.53
51.112.229.205

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

[droplet]
51.112.253.19

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

```

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible ec2 -i hosts -m ping
[WARNING]: Host '3.28.132.53' 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.
3.28.132.53 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

[WARNING]: Host '51.112.229.205' 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/ansi
ble-core/2.20/reference_appendices/interpreter_discovery.html for more information.
51.112.229.205 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible 3.28.132.53 -i hosts -m ping
[WARNING]: Host '3.28.132.53' 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/ansi
ble-core/2.20/reference_appendices/interpreter_discovery.html for more information.
3.28.132.53 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible droplet -i hosts -m ping
[WARNING]: Host '51.112.253.19' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installati
on of another Python interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/ansi
ble-core/2.20/reference_appendices/interpreter_discovery.html for more information.
51.112.253.19 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '51.112.253.19' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cau
se a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
51.112.253.19 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

[WARNING]: Host '3.28.132.53' 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.
3.28.132.53 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

[WARNING]: Host '51.112.229.205' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could ca
use a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
51.112.229.205 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}

```

Task 4 – Global ansible.cfg & first nginx playbook



```
Urwa@Urwa-OptiPlex-5090: ~ % cd .\workspaces\terraform_machine\main> cat hosts
[ec2]
3.28.132.53
51.112.229.205

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

[droplet]
51.112.253.19

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

```
● Urwaa012 → /workspaces/terraform_machine (main) $ ansible all -i hosts -m ping
[WARNING]: Host '3.28.132.53' 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.
3.28.132.53 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '51.112.229.205' 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.
51.112.229.205 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host '51.112.253.19' 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.
51.112.253.19 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ touch my-playbook.yaml
● @Urwa012 → /workspaces/terraform_machine (main) $ ls -la my-playbook.yaml
-rw-rw-rw- 1 codespace codespace 0 Jan 10 18:55 my-playbook.yaml
```

```
GNU nano 7.2                                     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
```

```

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.132.53' 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: [3.28.132.53]
[WARNING]: Host '51.112.229.205' 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: [51.112.229.205]

TASK [install nginx and update cache] ****
changed: [51.112.229.205]
changed: [3.28.132.53]

TASK [start nginx server] ****
changed: [51.112.229.205]
changed: [3.28.132.53]

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

```

← → ⌂ △ Not secure 3.28.132.53

Welcome to nginx!

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

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

Thank you for using nginx.

```

GNU nano 7.2                                     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

```

```

● @Urwa012 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '51.112.253.19' 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: [51.112.253.19]

TASK [install nginx and update cache] ****
changed: [51.112.253.19]

TASK [start nginx server] ****
changed: [51.112.253.19]

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

```



Task 5 – Single nginx target group & HTTPS prerequisites

```
● @Urwa012 → /workspaces/terraform_machine (main) $ touch ansible.cfg
● @Urwa012 → /workspaces/terraform_machine (main) $ ls -la ansible.cfg
-rw-rw-rw- 1 codespace codespace 0 Jan 10 20:00 ansible.cfg
```

```
EUF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat ansible.cfg
[defaults]
host_key_checking = False
interpreter_python = /usr/bin/python3
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
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
}

module.myapp-webserver[1].aws_security_group.web_sg: Destruction complete after 1s
module.myapp-webserver[2].aws_instance.myapp-server: Still destroying... [id=i-070a82ffb39ceec81, 01m00s elapsed]
module.myapp-webserver[2].aws_instance.myapp-server: Still destroying... [id=i-070a82ffb39ceec81, 01m10s elapsed]
module.myapp-webserver[2].aws_instance.myapp-server: Destruction complete after 1m11s
module.myapp-webserver[2].aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-2]
module.myapp-webserver[2].aws_security_group.web_sg: Destroying... [id=sg-035051cc8b8f16fb9]
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

Apply complete! Resources: 0 added, 1 changed, 6 destroyed.

Outputs:
webserver_public_ips = [
  "3.28.132.53",
]

● @Urwa012 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "3.28.132.53",
]
```

```

J
● @Urwa012 → /workspaces/terraform_machine (main) $ cat > hosts <<EOF
[nginx]
3.28.132.53

[nginx:vars]
ansible_ssh_private_key_file=~/ssh/id_ed25519
ansible_user=ec2-user
EOF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat hosts
[nginx]
3.28.132.53

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

```

```

● @Urwa012 → /workspaces/terraform_machine (main) $ cat my-playbook.yaml
---
- 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

```

```

● @Urwa012 → /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.cfg source. For more information see https://docs.ansible.com/ansible-devel/reference_appendices/config.html#cfg-in-world-writable-dir

PLAY [Configure nginx web server] ****
TASK [Gathering Facts] ****
[WARNING]: Host '3.28.132.53' 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: [3.28.132.53]

TASK [install nginx and update cache] ****
ok: [3.28.132.53]

TASK [install openssl] ****
ok: [3.28.132.53]

TASK [start nginx server] ****
changed: [3.28.132.53]

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

```

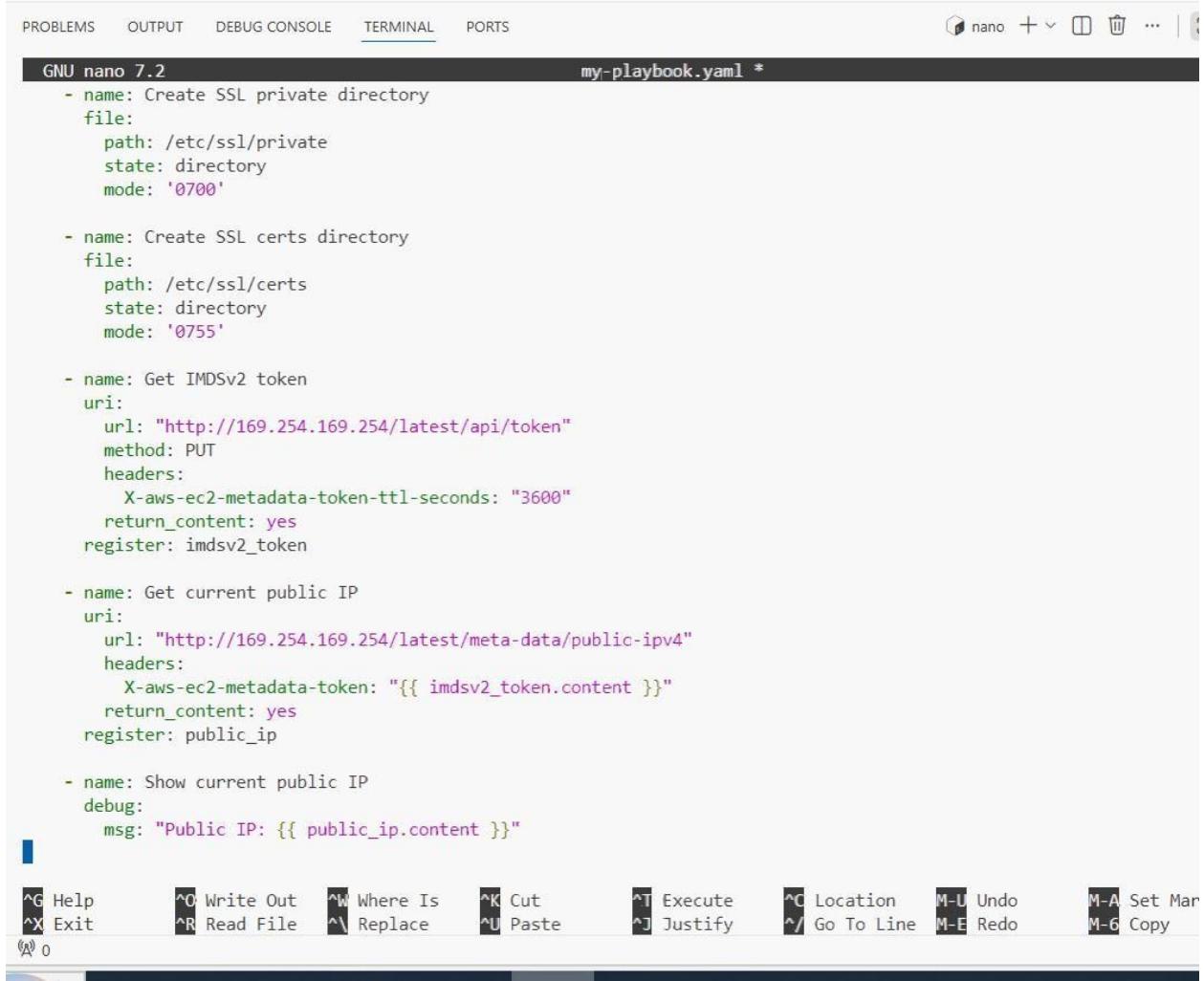
Welcome to nginx!

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

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

Thank you for using nginx!

Task 6 - Ansible-managed SSL certificates



The screenshot shows a terminal window with the title "my-playbook.yaml *". The window contains an Ansible YAML configuration file. The file defines several tasks:

- A task to create two SSL private directories: "/etc/ssl/private" and "/etc/ssl/certs". Both are set to be directories with mode 0700 and 0755 respectively.
- A task to get an IMDSv2 token using a PUT request to "http://169.254.169.254/latest/api/token". The response content is stored in the variable "imds_v2_token".
- A task to get the current public IP address using a GET request to "http://169.254.169.254/latest/meta-data/public-ipv4". The response content is stored in the variable "public_ip".
- A task to output the current public IP address to the terminal.

At the bottom of the terminal window, there is a menu bar with various keyboard shortcuts for file operations like Help, Exit, Write Out, Read File, etc., and application-specific commands like Cut, Paste, Execute, Location, Undo, Redo, Set Mar, and Copy.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
GNU nano 7.2                                         my-playbook.yaml *
- 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 }}"

^G Help      ^O Write Out   ^W Where Is   ^K Cut        ^T Execute   ^C Location   M-U Undo   M-A Set Mar
^X Exit     ^R Read File   ^\ Replace    ^U Paste     ^J Justify   ^/ Go To Line M-E Redo   M-C Copy
(2) 0
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ ansible-playbook -i hosts my-playbook.yaml
TASK [install openssl] ****
ok: [3.28.132.53]
TASK [start nginx server] ****
ok: [3.28.132.53]
PLAY [Configure SSL certificates] ****
TASK [Gathering Facts] ****
ok: [3.28.132.53]
TASK [Create SSL private directory] ****
changed: [3.28.132.53]
TASK [Create SSL certs directory] ****
changed: [3.28.132.53]
TASK [Get IMDSV2 token] ****
ok: [3.28.132.53]
TASK [Get current public IP] ****
ok: [3.28.132.53]
TASK [Show current public IP] ****
ok: [3.28.132.53] => {
    "msg": "Public IP: 3.28.132.53"
}
TASK [Generate self-signed SSL certificate] ****
changed: [3.28.132.53]
PLAY RECAP ****
3.28.132.53 : ok=11    changed=3      unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ ssh ec2-user@3.28.132.53 -i ~/.ssh/id_ed25519 "sudo cat /etc/ssl/certs/selfsigned.crt"
-----BEGIN CERTIFICATE-----
MIIDOzCCA0i0gAwIBAgIUTKYiAvkwXWDBWeeIb6lE8f12wS0wDQYJKoZIhvNAQEL
BQAwFjEUMBIGA1UEAwLMy4yOC4xhZIuNTNmHcNMjYwMTExMjAxOTAyWhcNMjcw
MTExMjAxOTAyJwJAWMRQwEgYDVQDDASzLjI4LjEZmI41MzCCASInDQYJKoZIhvCN
AQEBBAQDggTApDCCAQoCggEBAIrN9TqGcsL2Uk3AuVa5sFjjtJodK7h8x3IA9Ah
auhohskhHAMFaLkbTdWkImM49g7TbccQVZ+dulLaKTyEQs+VTz+/jz3n8NG6/JY1/
Mp8u09cqvZEt9LgTfGloDYPJNpQk3V4m+9wHSJdNhRsZ0OpEuHmj5Qk07Ss0+
j+eADg0BDbTfBwH215I20w8CojOPY0023HEFWfnbi2
imhELP1U2f4895/DehvyNjfZKF1PanL6mRL5F4E3xpahWdytaUJN1eJvvcb0i
XTR7Ec2TjqRFIzo50UDu0rNpo/3gEFs1ab9MjttrJdwQW2EXiHedlwQquoBf
cfgf17uo/HsNHkuJXZYNPz0xBz0a2nBU2wQjWmEfAonCw1bsImDZjPojyU3im
zaa4OBDbT6FV116r7LF+c1YPM22lm9yI4Uj+2Z2qrkPAVZ8yb+peLyTkfevaWtC1
RooKmhBaghBAEAEcggeABt3bRyE2ly50Jw0mnha2wArtdVd5PceEmxkiwszdIpSw
tpNPl/j3bcu1wx1Qde/RvsUEqsGsaRnw9mCslsh4cpxJx6r+huHZaX9SK5Da0VhQ
3+jCaJ2CsQw7W1/crUoRdp9awIRsmS+p66y5w2j3MTYR1uirn27Pcog111DUuD
HmldzME28A+A1wnfeODCeNKKHzssbDKA9qRnNa/hni3gd8+FhdAZfxAnfIMMP
ub+ihTdnW4y5nKEOrwd//e0bLc7tU6AeNeKhtOr-aTgY553AOJ456d5x6VmDx7
70WaR4QZQl9B3unopWRFp4L2zcceC9sc7fjs+r+BmQKBgQC+0Y3V7p+EM1NAVyrT
X//im/9rRz0MBQz5FDpwiukqqfSRSsBmxJswQRvx9F1Gd/1H1lhYDWN49jp71c
2f1SCi721IShW508LmzU0Ls9IMMHgw4uA02uPBV1khe/wgqxwRzIANGSVwoqLSs
ZTIIGBY1zWjWa1u7vp4DSqMLQkBqG6C/N/M/tcjq8CS3kdhnYEjyighf/EXZ1I
AZFLB0oxIXRNsyu/QE4aPk0MTwtv1aehJco9PmqtBqyh4Ypab52dYecl+ja3ETI
UgjmBkh5qADT/QYp18DoggfwJ+rv3zijm0sfvgVysSzx/FASjmTkczie9f6Lly/U7
DeR1jpnkZQKBgQcuUYV/MCOpmkfPeV1Y1p6TAR0Hmd634ksfLO4Ad2Tr/fz+Up/
GdfNJk9G5oYEsHSU0m+xE2sgxbTqVyx1ltsEHNV1/AvuAzua/0awnq133GtzQcf
0++plx2Hb043/aQuCv59hS4bINaakdlM6ytZydwDAX3EHnUNQilCaQKBgQC
bfNUTBRSsh+1MA1kj/o1CQ58BjN+emrNCL/86whm1UV1saz28C51xiwQlx2eMGr
NSOVII+qAz9y7PtbbQbcRHzVRDthat/ntdNx1mJUwCiggMiYqZ4dsA91i2Ev0
naApkrRjy1ThkwpvAbso/68+YcpqGXlqRJiJdb2aQKBgQC3H6gVq1j5+r7mJ0BW
sJ0WAMAU1Cd70Ut3vwAkGNM+f750MId7okaKw1ve4s1PBLoht7gPKhgVGPrJnRS
OfwJyoEJj1zTznoJK5eur8gVD1uHkNmMnTX3awfLevEcseSu7V3jQuRHRA6E98sh
6Fn4jDR/tjvSAFQKy+7XyQxcnw==
-----END PRIVATE KEY-----
```

Task 7 - PHP front-end deployment with templates

```

● @Urwa012 → /workspaces/terraform_machine (main) $ mkdir -p files templates
● @Urwa012 → /workspaces/terraform_machine (main) $ touch files/index.php
● @Urwa012 → /workspaces/terraform_machine (main) $ touch templates/nginx.conf.j2
● @Urwa012 → /workspaces/terraform_machine (main) $ 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

```

```

GNU nano 7.2                                         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);

```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A
 ^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6
 ^Q 0

```
GNU nano 7.2                               templates/nginx.conf.j2 *
```

```
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;
    }

    server {
        listen 443 ssl;
        server_name {{ server_public_ip }};
        ssl_certificate /etc/ssl/certs/selfsigned.crt;
    }
}
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set N
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy
^A 0

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

GNU nano 7.2 my-playbook.yaml *

```
- 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:
    name: nginx
    state: restarted

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

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^V Replace ^J Justify ^Y Go To Line M-E Redo
M-A Set Mark M-6 Copy

@Urwa012 → /workspaces/terraform_machine (main) \$ ansible-playbook -i hosts my-playbook.yaml

```
PLAY [Get current public IP] ****
ok: [3.28.132.53]

TASK [Show current public IP] ****
ok: [3.28.132.53] => {
    "msg": "Public IP: 3.28.132.53"
}

TASK [Generate self-signed SSL certificate] ****
ok: [3.28.132.53]

PLAY [Deploy Nginx website and configuration files] ****
TASK [Gathering Facts] ****
ok: [3.28.132.53]

TASK [Install php-fpm and php-curl] ****
changed: [3.28.132.53]

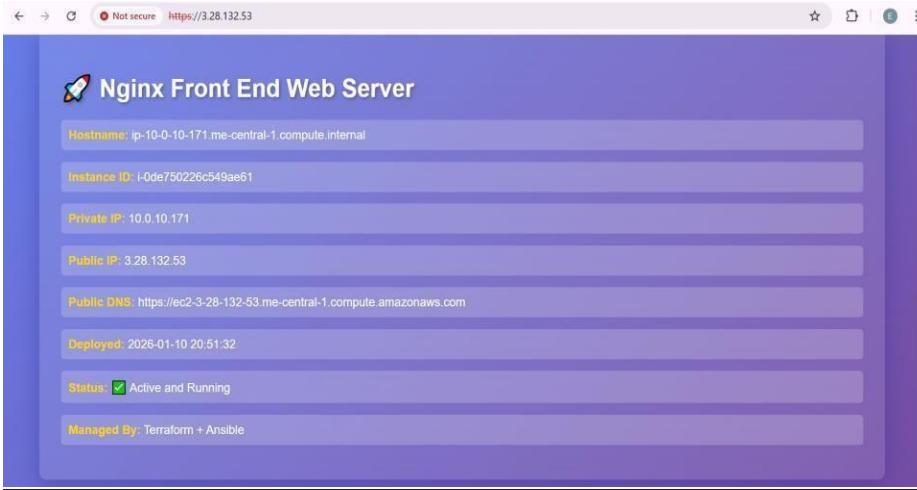
TASK [Copy website files] ****
changed: [3.28.132.53]

TASK [Copy nginx.conf template] ****
changed: [3.28.132.53]

TASK [Restart nginx] ****
changed: [3.28.132.53]

TASK [Start and enable php-fpm] ****
changed: [3.28.132.53]

PLAY RECAP ****
3.28.132.53 : ok=17    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



Task 8 – Docker & Docker Compose provisioning via Ansible

```
[...] -> null
module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-041c0cebfd69ac195]
module.myapp-webserver[0].aws_instance.myapp-server: Destroying... [id=i-0de750226c549ae61]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id=igw-05f020657b7d70642]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-05f020657b7d70642, 00m10s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-05f020657b7d70642, 00m20s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 00m30s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-05f020657b7d70642, 00m30s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-05f020657b7d70642, 00m40s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 47s
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 00m50s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-0de750226c549ae61, 01m00s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Destruction complete after 1m0s
module.myapp-webserver[0].aws_key_pair.ssh-key: Destroying... [id=dev-serverkey-0]
module.myapp-subnet.aws_subnet.myapp_subnet_1: Destroying... [id=subnet-0a0c190af2d712a41]
module.myapp-webserver[0].aws_security_group.web_sg: Destroying... [id=sg-071d58cfb531ff9ab]
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-0b47c95f0961cc901]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 7 destroyed.
-
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 0s [id=dev-serverkey-0]
aws_vpc.myapp_vpc: Still creating... [00m10s elapsed]
aws_vpc.myapp_vpc: Creation complete after 12s [id=vpc-06dfda92e1f3495b6]
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 1s [id=igw-095f5cac643a8eb2f]
module.myapp-subnet.aws_default_route_table.main_rt: Creating...
module.myapp-subnet.aws_default_route_table.main_rt: Creation complete after 0s [id=rtb-0bbf30528f5643454]
module.myapp-webserver[0].aws_security_group.web_sg: Creation complete after 3s [id=sg-0b7be24da9c0ae129]
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-0ef895511c536f2b5]
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-0d0b8a07d04790590]

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

Outputs:
webserver_public_ips = [
    "158.252.35.247",
]
```

```
J
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform output
webserver_public_ips = [
  "158.252.35.247",
]
```

```
EOF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat hosts
[docker_servers]
158.252.35.247

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

```
@Urwa012 → /workspaces/terraform_machine (main) $ cat my-playbook.yaml
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') }}.tar.gz
      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: 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
```

```

@Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i hosts my-playbook.yaml
  warning: This connection is using the deprecated SSH protocol (SSH-2). Consider upgrading to SSH-3 to
  warning: prevent another Python interpreter could cause a different interpreter to be discovered. See https://docs.ansible.com/
  warning: ansible-core/2.20/reference_appendices/interpreter_discovery.html for more information.
ok: [158.252.35.247]

TASK [install docker and update cache] ****
changed: [158.252.35.247]

PLAY [Install Docker Compose] ****
TASK [Gathering Facts] ****
ok: [158.252.35.247]

TASK [create docker cli-plugins directory] ****
changed: [158.252.35.247]

TASK [install docker-compose] ****
changed: [158.252.35.247]

TASK [View architecture of the system] ****
ok: [158.252.35.247] => {
    "msg": "System architecture of 158.252.35.247 is x86_64"
}

TASK [Alternate method to view architecture of the system] ****
ok: [158.252.35.247] => {
    "msg": "System architecture of 158.252.35.247 is x86_64"
}

TASK [restart docker service] ****
changed: [158.252.35.247]

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

```

```

@Urwa012 → /workspaces/terraform_machine (main) $ ssh -o StrictHostKeyChecking=no ec2-user@158.252.35.247 -i ~/.ssh/id_ec25519 "sudo docker ps"
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES

```

Task 9 – Gitea Docker stack via Ansible + Terraform security group update

```

GNU nano 7.2                                     my-playbook.yaml *
- name: restart docker service
  service:
    name: docker
    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

  ^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location
  ^X Exit   ^R Read File   ^V Replace   ^U Paste   ^J Justify   ^/ Go To Lin
  ^A 0

```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ cat > project-vars.yaml <<EOF
normal_user: ec2-user
docker_compose_file_location: .
EOF
● @Urwa012 → /workspaces/terraform_machine (main) $ cat project-vars.yaml
normal_user: ec2-user
docker_compose_file_location: .
```

```
GNU nano 7.2                                my-playbook.yaml *
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

- name: Deploy Docker Containers
  hosts: all
  become: true
  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"
      changed_when: "'Creating' in compose_result.stdout or 'Recreating' in compose_result.stderr"

^G Help      ^C Write Out   ^W Where Is     ^K Cut          ^T Execute      ^C Location    M-U Undo
^X Exit      ^R Read File   ^\ Replace      ^U Paste         ^J Justify      ^/ Go To Line  M-E Redo
^Q n          ^O Open File   ^I Insert       ^P Delete        ^L Select       M-A Select All M-G
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ cat compose.yaml
- 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
expose:
- 5432
networks:
- webnet

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

networks:
webnet:
  name: webnet
```

```

@Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -i hosts my-playbook.yaml

TASK [add user to docker group] *****
changed: [158.252.35.247]

TASK [reconnect to apply group changes] *****
ok: [158.252.35.247]

TASK [verify docker access] *****
ok: [158.252.35.247]

TASK [display docker ps output] *****
ok: [158.252.35.247] => {
    "docker_ps.stdout": "CONTAINER ID     IMAGE     COMMAND     CREATED     STATUS     PORTS     NAMES"
}

TASK [fail if docker is not accessible] *****
skipping: [158.252.35.247]

PLAY [Deploy Docker Containers] *****

TASK [Gathering Facts] *****
ok: [158.252.35.247]

TASK [check if docker-compose file exists] *****
ok: [158.252.35.247]

TASK [copy docker-compose file] *****
changed: [158.252.35.247]

TASK [deploy containers using docker-compose] *****
ok: [158.252.35.247]

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

```

```

GNU nano 7.2                                     modules/webserver/main.tf *

  protocol      = "tcp"
  cidr_blocks  = [var.my_ip]
}
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"]
}
egress {
  from_port    = 0
  to_port      = 0
  protocol     = "-1"
  cidr_blocks  = ["0.0.0.0/0"]
  prefix_list_ids = []
}
ingress {
  from_port    = 3000
  to_port      = 3000
  protocol     = "tcp"
  cidr_blocks  = ["0.0.0.0/0"]
}
tags = {
  Name = "${var.env_prefix}-default-sg"
}

```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
 ^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

```

        + security_groups  = []
        + self             = false
        + to_port          = 22
    },
]
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-0b7be24da9c0ae129]
module.myapp-webserver[0].aws_security_group.web_sg: Modifications complete after 1s [id=sg-0b7be24da9c0ae129]

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

Outputs:

webserver_public_ips = [
    "158.252.35.247",
]

```

Not secure 158.252.35.247:3000

Zaheena

Issues Pull Requests Milestones Explore

Account was successfully created. Welcome!

No Activity

You are currently not following any repositories or users, so there is no content to display. You can explore repositories or users of interest from the links below.

Repositories

There are no repositories yet.

Task 10 – Automating Ansible with Terraform (null resource)

```

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 '[docke'
  }
}

```

Help Write Out Where Is Cut Execute Location Undo Set Mark
 Exit Read File Replace Paste Justify Go To Line Redo Copy

```

J

Plan: 0 to add, 0 to change, 8 to destroy.

Changes to Outputs:
- webserver_public_ips = [
  - "3.28.74.17",
] -> null

null_resource.configure_server: Destroying... [id=7568289974248137620]
null_resource.configure_server: Destruction complete after 0s
module.myapp-subnet.aws_default_route_table.main_rt: Destroying... [id=rtb-068b2f3502fb7129b]
module.myapp-subnet.aws_default_route_table.main_rt: Destruction complete after 0s
module.myapp-webserver[0].aws_instance.myapp-server: Destroying... [id=i-03d076caa5aec5613]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destroying... [id=igw-0e09009054714e425]
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-03d076caa5aec5613, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e09009054714e425, 00m10s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e09009054714e425, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Still destroying... [id=igw-0e09009054714e425, 00m20s elapsed]
module.myapp-subnet.aws_internet_gateway.myapp_igw: Destruction complete after 27s
module.myapp-webserver[0].aws_instance.myapp-server: Still destroying... [id=i-03d076caa5aec5613, 00m30s elapsed]
module.myapp-webserver[0].aws_instance.myapp-server: Destruction complete after 30s
module.myapp-subnet.myapp_subnet_1: Destroying... [id=subnet-01faa92ca7723d79e]
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-02b2bb755f03f31b9]
module.myapp-subnet.myapp_subnet_1: Destruction complete after 0s
module.myapp-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-0545ed0ce584d49bb]
aws_vpc.myapp_vpc: Destruction complete after 1s

Destroy complete! Resources: 8 destroyed.

```

```

@Urwa012 ~ /workspaces/terraform_machine (main) $ terraform apply -auto-approve
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: [40.172.221.203]

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

null_resource.configure_server (local-exec): ok: [40.172.221.203]

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

null_resource.configure_server (local-exec): changed: [40.172.221.203]

null_resource.configure_server (local-exec): TASK [deploy containers using docker-compose] ****
****

null_resource.configure_server: Still creating... [01m50s elapsed]
null_resource.configure_server: Still creating... [02m00s elapsed]
null_resource.configure_server (local-exec): ok: [40.172.221.203]

null_resource.configure_server (local-exec): PLAY RECAP ****
****

null_resource.configure_server (local-exec): 40.172.221.203 : ok=13    changed=6    unreachable=0
      skipped=0   rescued=0   ignored=0

null_resource.configure_server: Creation complete after 2m6s [id=595861089165142826]

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

Outputs:

webserver_public_ips = [
  "40.172.221.203",
]

```

```

GNU nano 7.2                                     my-playbook.yaml *
- name: Wait for connection
  hosts: all
  gather_facts: no
  tasks:
    - name: Wait for SSH to come up
      wait_for_connection:
        delay: 10
        timeout: 300

    ---

    - 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:

```

^G Help ^O Write Out ^W Where Is ^X Cut ^A Execute ^C Location M-L Undo M-A Set Mark
 ^X Exit ^R Read File ^W Replace ^U Paste ^D Justify ^V Go To Line M-E Redo M-G Copy
 ^Q 0

```

null_resource.configure_server (local-exec): changed: [40.172.221.203]

null_resource.configure_server (local-exec): TASK [deploy containers using docker-compose]
 ****
null_resource.configure_server: Still creating... [01m50s elapsed]
null_resource.configure_server: Still creating... [02m00s elapsed]
null_resource.configure_server (local-exec): ok: [40.172.221.203]

null_resource.configure_server (local-exec): PLAY RECAP ****
null_resource.configure_server (local-exec): 40.172.221.203 : ok=13    changed=0
      skipped=0    rescued=0    ignored=0

null_resource.configure_server: Creation complete after 2m6s [id=595861089165142826]

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

Outputs:

webserver_public_ips = [
  "40.172.221.203",
]
```

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 that this user must have access to the repository root path.

Server Domain * 40.172.221.203
Domain or host address for the server.

SSH Server Port 22
Port number your SSH server listens on. Leave empty to disable.

Gitea HTTP Listen Port * 8000
Port number the Gitea web server will listen on.

Gitea Base URL * http://40.172.221.203:8000/
Base address for HTTPS clone URLs and email notifications.

Log Path * /data/gitea/log
Log files will be written to this directory.

Enable Update Checker
Checks for new version releases periodically by connecting to gitea.io.

Optional Settings

- Email Settings
- Server and Third Party Service Settings
- Administrator Account Settings

These configuration options will be written into: /data/gitea/conf/app.ini

Install Gitea

Task 11 – Dynamic inventory with aws_ec2 plugin

```
● @Urwa012 → /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
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ touch inventory_aws_ec2.yaml
ls -la inventory_aws_ec2.yaml
-rw-rw-rw- 1 codespace codespace 0 Jan 10 22:04 inventory_aws_ec2.yaml
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ cat inventory_aws_ec2.yaml
---
plugin: aws_ec2
regions:
- me-central-1
```

The screenshot shows a terminal session with three code snippets. The first snippet shows the creation of an Ansible configuration file `ansible.cfg` with the `aws_ec2` plugin enabled. The second snippet shows the creation of a dynamic inventory file `inventory_aws_ec2.yaml`. The third snippet shows the contents of a `main.tf` file being edited in a nano text editor. The `main.tf` file defines two modules: `myapp-webserver` and `myapp-webserver-prod`, each with specific AWS parameters like instance type, availability zone, and VPC connections.

```
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
}

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
}

resource "null_resource" "configure_server" {
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Loca
^X Exit ^R Read File ^V Replace ^U Paste ^J Justify ^Y Go T
^Q 0

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
GNU nano 7.2                                         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]
}

@Urwa012 → /workspaces/terraform_machine (main) $ terraform init
terraform apply -auto-approve
+ revoke_rules_on_delete = false
+ tags = {
  + "Name" = "prod-default-sg"
}
+ tags_all = {
  + "Name" = "prod-default-sg"
}
+ vpc_id = "vpc-06524a1aad9a6dd7b"

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 0s [id=prod-serverkey-0]
module.myapp-webserver-prod[0].aws_security_group.web_sg: Creation complete after 3s [id=sg-0528db41afc47acee]
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-01952f53748e45806]

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

Outputs:

prod-webserver_public_ips = [
  "40.172.221.121",
]
webserver_public_ips = [
  "40.172.221.203",
]
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ terraform output
prod-webserver_public_ips = [
  "40.172.221.121",
]
webserver_public_ips = [
  "40.172.221.203",
]
```

```
|--@ungrouped:  
● @Urwa012 → /workspaces/terraform_machine (main) $ /usr/local/py-utils/venvs/ansible-core/bin/python -m pip install boto3  
botocore  
Collecting boto3  
  Using cached boto3-1.42.25-py3-none-any.whl.metadata (6.8 kB)  
Collecting botocore  
  Using cached botocore-1.42.25-py3-none-any.whl.metadata (5.9 kB)  
Collecting jmespath<2.0.0,>=0.7.1 (from boto3)  
  Using cached jmespath-1.0.1-py3-none-any.whl.metadata (7.6 kB)  
Collecting s3transfer<0.17.0,>=0.16.0 (from boto3)  
  Using cached s3transfer-0.16.0-py3-none-any.whl.metadata (1.7 kB)  
Collecting python-dateutil<3.0.0,>=2.1 (from botocore)  
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)  
Collecting urllib3!=2.2.0,<3,>=1.25.4 (from botocore)  
  Downloading urllib3-2.6.3-py3-none-any.whl.metadata (6.9 kB)  
Collecting six>=1.5 (from python-dateutil<3.0.0,>=2.1->botocore)  
  Downloading six-1.17.0-py2.py3-none-any.whl.metadata (1.7 kB)  
Using cached boto3-1.42.25-py3-none-any.whl (140 kB)  
Using cached botocore-1.42.25-py3-none-any.whl (14.6 MB)  
Using cached jmespath-1.0.1-py3-none-any.whl (20 kB)  
Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)  
Using cached s3transfer-0.16.0-py3-none-any.whl (86 kB)  
Downloading urllib3-2.6.3-py3-none-any.whl (131 kB)  
Downloading six-1.17.0-py2.py3-none-any.whl (11 kB)  
Installing collected packages: urllib3, six, jmespath, python-dateutil, botocore, s3transfer, boto3  
Successfully installed boto3-1.42.25 botocore-1.42.25 jmespath-1.0.1 python-dateutil-2.9.0.post0 s3transfer-0.16.0 six-1.17.0 urllib3-2.6.3
```

```
Successfully installed boto3-1.42.25 botocore-1.42.25 jmespath-1.0.1 python-dateutil-2.9.0.post0 s3transfer-0.16.0 six-1.17.0 urllib3-2.6.3  
● @Urwa012 → /workspaces/terraform_machine (main) $ $(which python) -c "import boto3, botocore; print(boto3.__version__)"  
1.42.25
```

```
@Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_CONFIG=./ansible.cfg ansible-inventory -i inventory_aws_ec2.yaml  
ml --graph  
@all:  
  |--@ungrouped:  
  |--aws_ec2:  
  |  |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com  
  |  |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com
```

Task 12 – Filtering EC2 instances by tags & instance type

```
EOF  
● @Urwa012 → /workspaces/terraform_machine (main) $ cat inventory_aws_ec2.yaml  
---  
plugin: aws_ec2  
regions:  
  - me-central-1  
keyed_groups:  
  - key: tags  
    prefix: tag  
    separator: "_"  
  
    separator: "_"  
● @Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_CONFIG=./ansible.cfg ansible-inventory -i inventory_aws_ec2.yaml  
ml --graph  
@all:  
  |--@ungrouped:  
  |--aws_ec2:  
  |  |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com  
  |  |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com  
  |--@tag_Name_prod_ec2_instance_0:  
  |  |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com  
  |--@tag_Name_dev_ec2_instance_0:  
  |  |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com
```

```
EUR
● @Urwa012 → /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: "_"
```

```
● @Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_CONFIG=./ansible.cfg ansible-inventory -i inventory_aws_ec2.yaml --graph
@all:
  |--@ungrouped:
  |--@aws_ec2:
  |   |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com
  |   |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com
  |--@tag_Name_prod_ec2_instance_0:
  |   |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com
  |--@instance_type_t3_nano:
  |   |--ec2-40-172-221-121.me-central-1.compute.amazonaws.com
  |--@tag_Name_dev_ec2_instance_0:
  |   |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com
  |--@instance_type_t3_micro:
  |   |--ec2-40-172-221-203.me-central-1.compute.amazonaws.com
```

```
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
            mode: '0700'

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

  ^G Help      ^O Write Out     ^W Where Is      ^K Cut
  ^X Exit      ^R Read File     ^L Replace      ^U Paste
  0  ↵ 0
```

```

@Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook \
-i inventory_aws_ec2.yaml \
-u ec2-user \
--private-key ~/.ssh/id_ed25519 \
my-playbook.yaml

PLAY [Deploy Nginx website and configuration files] ****
TASK [Gathering Facts] ****
ok: [ec2-40-172-221-121.compute.amazonaws.com]
ok: [ec2-40-172-221-203.compute.amazonaws.com]

TASK [install php-fpm and php-curl] ****
ok: [ec2-40-172-221-121.compute.amazonaws.com]
ok: [ec2-40-172-221-203.compute.amazonaws.com]

TASK [Copy website files] ****
ok: [ec2-40-172-221-121.compute.amazonaws.com]
ok: [ec2-40-172-221-203.compute.amazonaws.com]

TASK [Copy nginx.conf template] ****
ok: [ec2-40-172-221-121.compute.amazonaws.com]
ok: [ec2-40-172-221-203.compute.amazonaws.com]

TASK [Restart nginx] ****
changed: [ec2-40-172-221-203.compute.amazonaws.com]
changed: [ec2-40-172-221-121.compute.amazonaws.com]

TASK [Start and enable php-fpm] ****
ok: [ec2-40-172-221-121.compute.amazonaws.com]
ok: [ec2-40-172-221-203.compute.amazonaws.com]

PLAY RECAP ****
ec2-40-172-221-121.compute.amazonaws.com : ok=18    changed=1      unreachable=0      failed=0      skipped=0      rescued=0      ignored=0
ec2-40-172-221-203.compute.amazonaws.com : ok=18    changed=1      unreachable=0      failed=0      skipped=0      rescued=0      ignored=0

```

```

@Urwa012 → /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.compute.amazonaws.com] => {
  "msg": "Public IP: 40.172.221.203"
}

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

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

PLAY [Deploy Nginx website and configuration files] ****
TASK [Gathering Facts] ****
ok: [ec2-40-172-221-203.compute.amazonaws.com]

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

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

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

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

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

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

```

Urwa012 → /workspaces/terraform_machine (main) \$

```
@Urwa012 → /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
```

```
@Urwa012 → /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
```

```

@Urwa012 → /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

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
GNU nano 7.2                               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
inventory = ./inventory_aws_ec2.yaml

```

```

@Urwa012 → /workspaces/terraform_machine (main) $ ANSIBLE_HOST_KEY_CHECKING=False ansible-playbook -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

```

Task 13 – Ansible roles: nginx, ssl, webapp

```

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'
  }
}

```

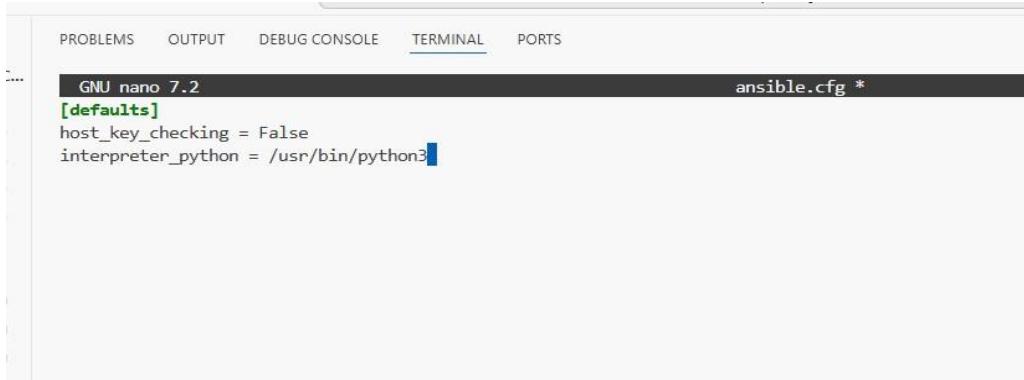
```

● @Urwa012 → /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:

```



The screenshot shows a terminal window with the following content:

```

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
```

```
@Urwa012 → /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
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

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
```

● @Urwa012 → /workspaces/terraform_machine/ansible (main) \$ ansible-playbook -i inventory/hosts my-playbook.yaml

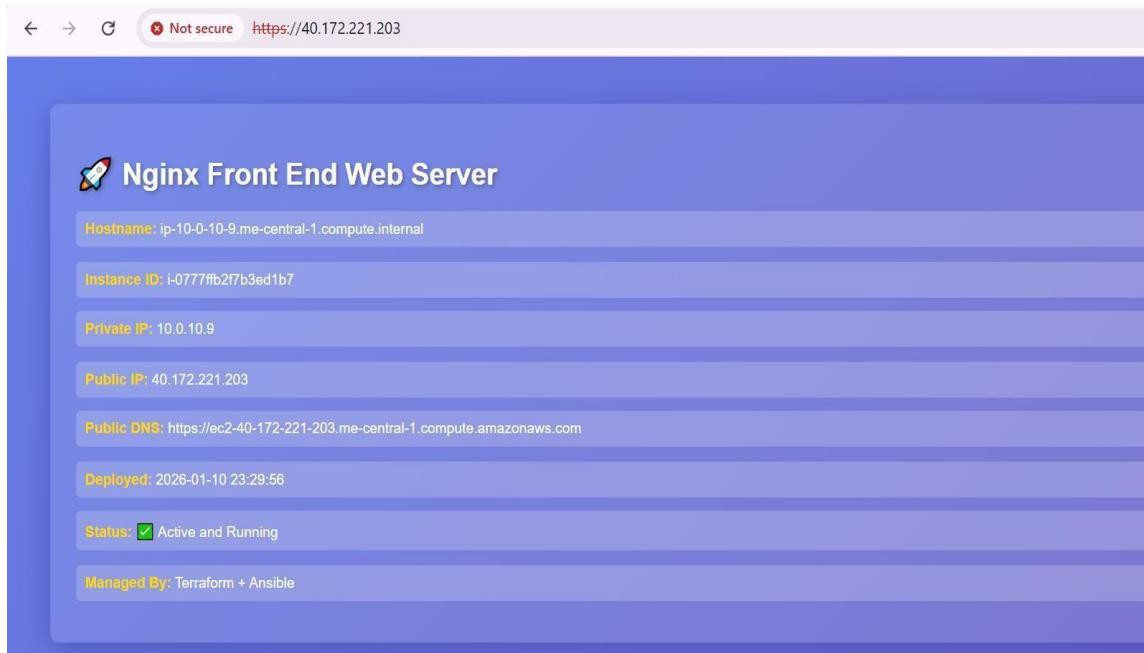
```
PLAY [Deploy NGINX Web Stack with SSL and PHP] ****
TASK [Gathering Facts] ****
ok: [40.172.221.203]

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
```



The screenshot shows a terminal window with the nano editor open. The file being edited is `roles/ssl/defaults/main.yml`. The content of the file is as follows:

```
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 }}"

^a Help      ^O Write Out   ^W Where Is   ^K Cut          ^T Execute   ^C Location   M-U Undo
^x Exit      ^R Read File   ^R Replace    ^U Paste        ^J Justify   ^G Go To Line M-E Redo
M-A Set Mar  M-G Copy
```

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, [
```

The screenshot shows a code editor interface with two tabs open. The top tab is titled 'roles/webapp/files/index.php *' and contains PHP code for fetching deployment information from AWS IMDS. The bottom tab is titled 'roles/webapp/handlers/main.yml *' and contains Ansible handler configuration for restarting services like nginx and php-fpm.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
```

```
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
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL FORMS

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;
    }
}
```

^G Help ^C Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-J T
^X Exit ^R Read File ^A Replace ^U Paste ^J Justify ^I Go To Line M-E Redo M-B Copy ^Q Wh

```
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
```

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   nano - ansi
GNU nano 7.2                                         my-playbook.yaml *
```

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

```
@Urwa012 → /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
```

The screenshot shows a web browser window with the following details:

- Address Bar:** Not secure https://40.172.221.203
- Page Title:** Nginx Front End Web Server
- Hostname:** ip-10-0-10-9.me-central-1.compute.internal
- Instance ID:** i-0777ffb2f7b3ed1b7
- 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

```

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

@Urwa012 → /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 complete! Resources: 0 destroyed.

```

```

● @Urwa012 → /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
}

```

