

MINI PROJECT: 1

TASKS:

1. Create an Ansible role to setup LAMP or XAMP
2. Create a Terraform script to provision stateful 3 tier architecture

TERRAFORM CONFIGURATION:

```
provider "aws" {
  region = "ap-south-1"
}

# Generate SSH Key
resource "tls_private_key" "ans_key" {
  algorithm = "RSA"
  rsa_bits  = 4096
}

resource "aws_key_pair" "ans_key" {
  key_name      = "ans_key"
  public_key    = tls_private_key.ans_key.public_key_openssh
}

resource "local_file" "private_key" {
  content  = tls_private_key.ans_key.private_key_pem
  filename = "ans_key1.pem"
}

# VPC
```

```
resource "aws_vpc" "new_vpc" {
  cidr_block = "10.0.0.0/16"
}

# Internet Gateway
resource "aws_internet_gateway" "igw" {
  vpc_id = aws_vpc.new_vpc.id
}

# Subnets
resource "aws_subnet" "public_subnet" {
  vpc_id            = aws_vpc.new_vpc.id
  cidr_block        = "10.0.1.0/24"
  availability_zone  = "ap-south-1a"
  map_public_ip_on_launch = true
  tags = {
    Name = "Public Subnet"
  }
}

resource "aws_subnet" "private_subnet" {
  vpc_id            = aws_vpc.new_vpc.id
  cidr_block        = "10.0.2.0/24"
  availability_zone  = "ap-south-1b"
  tags = {
    Name = "Private Subnet"
  }
}

# Public Route Table
resource "aws_route_table" "public_route_table" {
  vpc_id = aws_vpc.new_vpc.id
  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.igw.id
  }
}
```

```
}

resource "aws_route_table_association" "public_association" {
  subnet_id      = aws_subnet.public_subnet.id
  route_table_id = aws_route_table.public_route_table.id
}

# Security Group
resource "aws_security_group" "ansible_sg" {
  name          = "ansible_sg"
  vpc_id        = aws_vpc.new_vpc.id
  description   = "SG for Ansible allowing SSH, HTTP, HTTPS,
8080, ICMP"

  ingress {
    description = "allow ssh"
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
    description = "allow http"
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
    description = "allow https"
    from_port   = 443
    to_port     = 443
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }
}
```

```

}

ingress {
    description = "allow port 8080"
    from_port   = 8080
    to_port     = 8080
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
}

ingress {
    description = "allow all icmp"
    from_port   = -1
    to_port     = -1
    protocol    = "icmp"
    cidr_blocks = ["0.0.0.0/0"]
}

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

# EC2 Instances
resource "aws_instance" "ansible_server" {
    ami                     = "ami-0d0ad8bb301edb745"
    instance_type          = "t2.micro"
    subnet_id              = aws_subnet.public_subnet.id
    vpc_security_group_ids = [aws_security_group.ansible_sg.id]
    key_name               = aws_key_pair.ans_key.key_name
    associate_public_ip_address = true
}

```

```

    tags = {
        Name = "Ansible Control Server"
    }
}

resource "aws_instance" "managerial_node" {
    ami                = "ami-0d0ad8bb301edb745"
    instance_type      = "t2.micro"
    subnet_id          = aws_subnet.public_subnet.id
    vpc_security_group_ids = 
[aws_security_group.ansible_sg.id]
    key_name           = aws_key_pair.ans_key.key_name
    associate_public_ip_address = true
    tags = {
        Name = "Managerial Node 1"
    }
}

resource "aws_instance" "managerial_node2" {
    ami                = "ami-0d0ad8bb301edb745"
    instance_type      = "t2.micro"
    subnet_id          = aws_subnet.public_subnet.id
    vpc_security_group_ids = [aws_security_group.ansible_sg.id]
    key_name           = aws_key_pair.ans_key.key_name
    associate_public_ip_address = true
    tags = {
        Name = "Managerial Node 2"
    }
}

```

```
main.tf x
3-tier-architecture > main.tf > ...
133 resource "aws_instance" "managerial_node" {
134     vpc_security_group_ids = [aws_security_group.ansible_sg.id]
138     key_name               = aws_key_pair.ans_key.key_name
139     associate_public_ip_address = true
140     tags = {
141         Name = "Managerial Node 1"
142     }
143 }
144
145 resource "aws_instance" "managerial_node2" {
146     ami             = "ami-0d0ad8bb301edb745"
147     instance_type   = "t2.micro"
148     subnet_id       = aws_subnet.public_subnet.id
149     vpc_security_group_ids = [aws_security_group.ansible_sg.id]
150     key_name        = aws_key_pair.ans_key.key_name
151     associate_public_ip_address = true
152     tags = {
153         Name = "Managerial Node 2"
154     }
155 }
156
157
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS CODE REFERENCE LOG

aws_instance.managerial_node: Still creating... [00m10s elapsed]
aws_instance.ansible_server: Creation complete after 17s [id=i-000a16ff9d42e6e29]
aws_instance.managerial_node2: Creation complete after 17s [id=i-028b519d397943310]
aws_instance.managerial_node: Still creating... [00m20s elapsed]
aws_instance.managerial_node: Still creating... [00m30s elapsed]
aws_instance.managerial_node: Creation complete after 33s [id=i-07c025fb3199fe3c3]

Apply complete! Resources: 13 added, 0 changed, 0 destroyed.
PS D:\Terraform\3-tier-architecture>

Ln 157, Col 1 Spaces: 2 UTF-8 CRLF (Terraform Prettier

ANSIBLE CONFIGURATION:

aws

Search

[Alt+S]

Asia Pacific (Mumbai)

Rajesh S

```
visudo: /etc/sudoers.tmp unchanged
[root@Ansible-Control-Node ~]# su - Ansadmin
[Ansadmin@Ansible-Control-Node ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/Ansadmin/.ssh/id_rsa):
Created directory '/home/Ansadmin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/Ansadmin/.ssh/id_rsa
Your public key has been saved in /home/Ansadmin/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+ac1g8WZ3JS/TBmOfnaQ4YH2dftuXFjBX9K8UJaaQk8 Ansadmin@Ansible-Control-Node
The key's randomart image is:
+---[RSA 3072]-----+
|      o=.|
|      .Eo++|
|      .B B.B|
|      + B X O=|
|      S * + X+ |
|      + . o.o||
|      . + + =.+|
|      * o oo|
|      . ..|
+---[SHA256]-----+
[Ansadmin@Ansible-Control-Node ~]$
[Ansadmin@Ansible-Control-Node ~]$ cd .ssh
[Ansadmin@Ansible-Control-Node .ssh]$ ls
id_rsa id_rsa.pub
[Ansadmin@Ansible-Control-Node .ssh]$ exit
logout
[root@Ansible-Control-Node ~]#
```

i-000a16ff9d42e6e29 (Ansible Control Server)
PublicIPs: 65.0.98.53 PrivateIPs: 10.0.1.133

aws

Search

[Alt+S]

Asia Pacific (Mumbai)



Rajesh S

```
+ o oo|
|      ..|
+---[SHA256]-----+
[Ansadmin@Ansible-Control-Node ~]$
[Ansadmin@Ansible-Control-Node ~]$ cd .ssh
[Ansadmin@Ansible-Control-Node .ssh]$ ls
id_rsa id_rsa.pub
[Ansadmin@Ansible-Control-Node .ssh]$ exit
logout
[root@Ansible-Control-Node ~]# vim /etc/ssh/sshd_config
[root@Ansible-Control-Node ~]# systemctl restart sshd
[root@Ansible-Control-Node ~]# sudo mkdir -p /etc/ansible
[root@Ansible-Control-Node ~]# nano /etc/ansible/ansible.cfg
[root@Ansible-Control-Node ~]# ansible-config dump --only-changed
CONFIG FILE() = /etc/ansible/ansible.cfg
DEFAULT_BECOME(/etc/ansible/ansible.cfg) = True
DEFAULT_BECOME_ASK_PASS(/etc/ansible/ansible.cfg) = False
DEFAULT_BECOME_METHOD(/etc/ansible/ansible.cfg) = sudo
DEFAULT_BECOME_USER(/etc/ansible/ansible.cfg) = root
DEFAULT_FORKS(/etc/ansible/ansible.cfg) = 10
DEFAULT_GATHERING(/etc/ansible/ansible.cfg) = smart
DEFAULT_HOST_LIST(/etc/ansible/ansible.cfg) = ['/etc/ansible/hosts']
DEFAULT_LOG_PATH(/etc/ansible/ansible.cfg) = /root/.ansible.log
DEFAULT_PRIVATE_KEY_FILE(/etc/ansible/ansible.cfg) = /root/.ssh/id_rsa
DEFAULT_REMOTE_USER(/etc/ansible/ansible.cfg) = ansadmin
DEFAULT_STDOUT_CALLBACK(/etc/ansible/ansible.cfg) = yaml
DEFAULT_TIMEOUT(/etc/ansible/ansible.cfg) = 60
DEPRECATION_WARNINGS(/etc/ansible/ansible.cfg) = False
HOST_KEY_CHECKING(/etc/ansible/ansible.cfg) = False
RETRY_FILES_ENABLED(/etc/ansible/ansible.cfg) = False
[root@Ansible-Control-Node ~]#
```





i-000a16ff9d42e6e29 (Ansible Control Server)
PublicIPs: 65.0.98.53 PrivateIPs: 10.0.1.133

```
aws [Alt+S]
Q Search
Asia Pacific (Mumbai)
Rajesh S
[Ansadmin@Ansible-Control-Node ~]$ sudo vi /etc/ansible/lamp.yml
[Ansadmin@Ansible-Control-Node ~]$ cd /etc/ansible
[Ansadmin@Ansible-Control-Node ansible]$ ls
ansible.cfg  hosts  lamp.yml
[Ansadmin@Ansible-Control-Node ansible]$ cat lamp.yml
---
- name: Setup LAMP Stack on Managerial Nodes
  hosts: managers
  become: yes
  tasks:
    - name: Install Apache
      yum:
        name: httpd
        state: present
    - name: Start and Enable Apache
      service:
        name: httpd
        state: started
        enabled: yes
    - name: Install MariaDB (MySQL)
      yum:
        name: mariadb-server
        state: present
    - name: Start and Enable MariaDB
      service:
        name: mariadb
        state: started
```


MANAGERIAL NODE :



[Alt+S]



Asia Pacific (Mumbai) ▾



Rajesh S ▾

```
yum.noarch                                4.14.0-1.amzn2023.0.6                @System
zip.x86_64                               3.0-28.amzn2023.0.2                  @System
zlib.x86_64                               1.2.11-33.amzn2023.0.5               @System
zram-generator.x86_64                     1.1.2-70.amzn2023.0.1               @System
zram-generator-defaults.noarch             1.1.2-70.amzn2023.0.1               @System
zstd.x86_64                               1.5.5-1.amzn2023.0.1                @System





[Ansadmin@Ansible-Manerial-Node-1 ~]$ rpm -qa --last
php8.4-xml-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:58 2025
php8.4-sodium-8.4.10-1.amzn2023.0.1.x86_64  Sun Aug  3 07:14:58 2025
php8.4-mysqld-8.4.10-1.amzn2023.0.1.x86_64  Sun Aug  3 07:14:58 2025
php8.4-8.4.10-1.amzn2023.0.1.x86_64        Sun Aug  3 07:14:58 2025
libxslt-1.1.43-1.amzn2023.0.1.x86_64       Sun Aug  3 07:14:58 2025
libsodium-1.0.19-4.amzn2023.x86_64        Sun Aug  3 07:14:58 2025
php8.4-process-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-pdo-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:57 2025
php8.4-opcache-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-mbstring-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-fpm-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:57 2025
php8.4-common-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-cli-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:57 2025
nginxfilesystem-1.28.0-1.amzn2023.0.1.noarch Sun Aug  3 07:14:57 2025
mariadb105-server-utils-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:50 2025
mariadb105-server-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:50 2025
mariadb105-gssapi-server-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:49 2025
mariadb105-cracklib-password-check-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:49 2025
mariadb105-backup-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:49 2025
mariadb105-10.5.29-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:49 2025
perl-base-2.27-477.amzn2023.0.7.noarch      Sun Aug  3 07:14:37 2025
perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64 Sun Aug  3 07:14:37 2025
perl-Math-Complex-1.59-477.amzn2023.0.7.noarch Sun Aug  3 07:14:37 2025
perl-Math-BigRat-0.2624-500.amzn2023.0.2.noarch Sun Aug  3 07:14:37 2025
```

i-07c025fb3199fe3c3 (Managerial Node 1)

PublicIPs: 3.109.48.225 PrivateIPs: 10.0.1.125



[Alt+S]



Asia Pacific (Mumbai) ▾

Rajesh S ▾

```
[ec2-user@Ansible-Manerial-Node-2 ~]$ sudo su - Ansadmin
Last login: Sun Aug  3 06:41:44 UTC 2025 from 10.0.1.133 on pts/2
[Ansadmin@Ansible-Manerial-Node-2 ~]$ rpm -qa --last
php8.4-xml-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:58 2025
php8.4-sodium-8.4.10-1.amzn2023.0.1.x86_64  Sun Aug  3 07:14:58 2025
php8.4-mysqld-8.4.10-1.amzn2023.0.1.x86_64  Sun Aug  3 07:14:58 2025
php8.4-fpm-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:58 2025
php8.4-8.4.10-1.amzn2023.0.1.x86_64        Sun Aug  3 07:14:58 2025
libxslt-1.1.43-1.amzn2023.0.1.x86_64       Sun Aug  3 07:14:58 2025
libsodium-1.0.19-4.amzn2023.x86_64        Sun Aug  3 07:14:58 2025
php8.4-process-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-pdo-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:57 2025
php8.4-opcache-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-mbstring-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-common-8.4.10-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:57 2025
php8.4-cli-8.4.10-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:57 2025
nginxfilesystem-1.28.0-1.amzn2023.0.1.noarch Sun Aug  3 07:14:57 2025
mariadb105-server-utils-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:49 2025
mariadb105-server-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:49 2025
mariadb105-gssapi-server-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:48 2025
mariadb105-cracklib-password-check-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:48 2025
mariadb105-backup-10.5.29-1.amzn2023.0.1.x86_64 Sun Aug  3 07:14:48 2025
mariadb105-10.5.29-1.amzn2023.0.1.x86_64    Sun Aug  3 07:14:48 2025
perl-base-2.27-477.amzn2023.0.7.noarch      Sun Aug  3 07:14:37 2025
perl-Sys-Hostname-1.23-477.amzn2023.0.7.x86_64 Sun Aug  3 07:14:37 2025
perl-Math-Complex-1.59-477.amzn2023.0.7.noarch Sun Aug  3 07:14:37 2025
perl-Math-BigRat-0.2624-500.amzn2023.0.2.noarch Sun Aug  3 07:14:37 2025
perl-Math-BigInt-1.9998.39-2.amzn2023.0.2.noarch Sun Aug  3 07:14:37 2025
perl-FileHandle-2.03-477.amzn2023.0.7.noarch Sun Aug  3 07:14:37 2025
perl-File-Copy-2.34-477.amzn2023.0.7.noarch Sun Aug  3 07:14:37 2025
perl-Data-Dumper-2.174-460.amzn2023.0.2.x86_64 Sun Aug  3 07:14:37 2025
```

i-028b519d397943310 (Managerial Node 2)

PublicIPs: 13.235.45.149 PrivateIPs: 10.0.1.39