Automating a Multi-Tier Application on VMware with Ansible

Objective:

Using Ansible to fully automate the deployment, configuration, and orchestration of a multi-tier application (e.g., LAMP stack) across multiple CentOS VMs (master node, node1, and node2) in a VMware environment.

Steps:

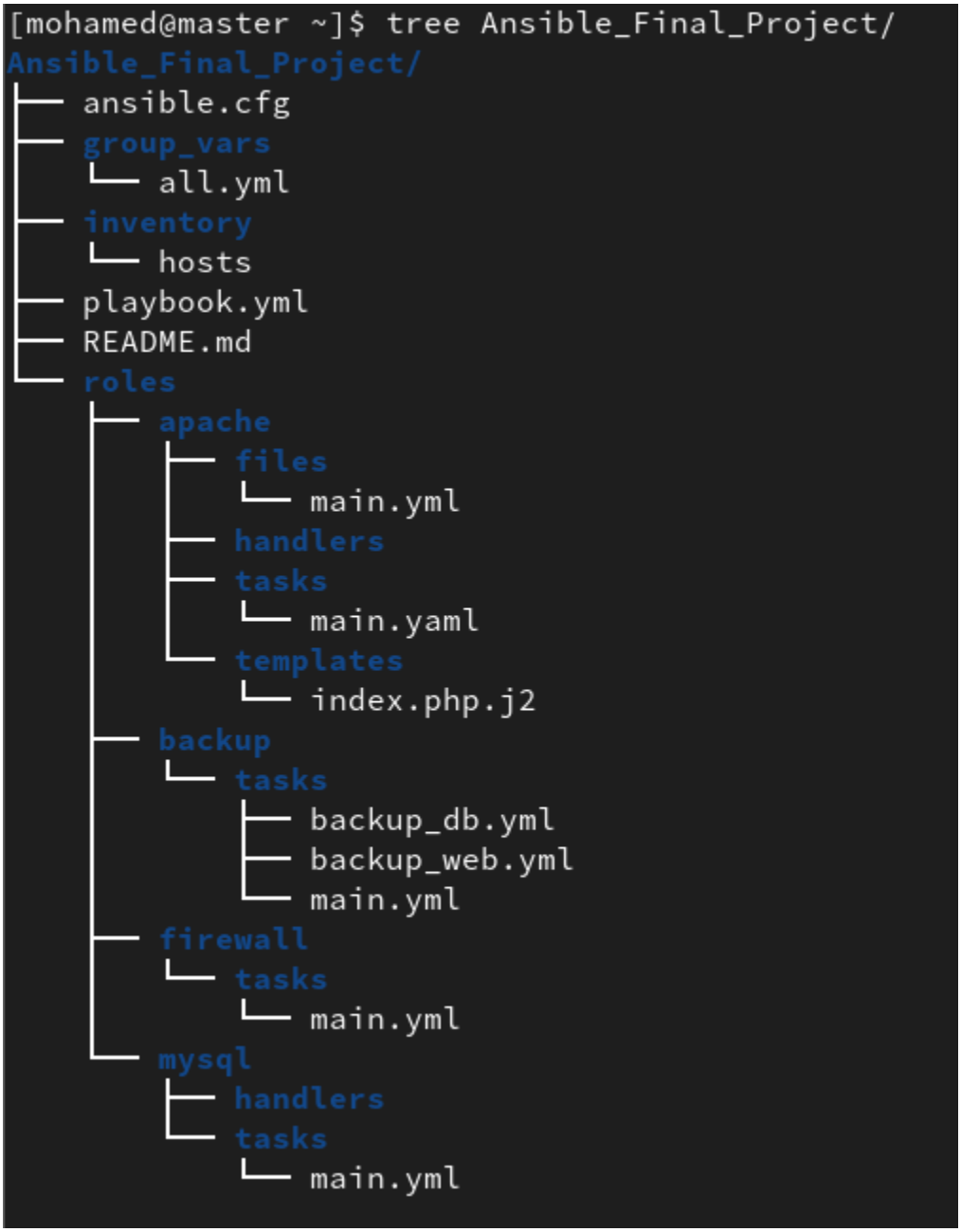
1. **Provisioning VMs**:
    - Use Ansible to deploy and configure two nodes (node1 and node2) on VMware.
    - Automate the installation of necessary dependencies like Apache on one node and MySQL on the other.
2. **Role-based Deployment**:
    - Use **Ansible roles** to separate the tasks:
        - **Web server role** (for node1 to host Apache/PHP).
        - **Database server role** (for node2 to host MySQL).
    - Deploy a simple PHP application on the web server node and configure it to communicate with the database on node2.
3. **Network Configuration & Load Balancing**:
    - Automate firewall rules, IP tables, and networking configurations between the nodes.
4. **Backup and Recovery Automation**:
    - Automate backups of the database on node2 using Ansible scheduled jobs.
    - Automate web content backup on node1.
5. **Security**:
    - Implement Ansible Vault to manage secrets such as database credentials.
    - Use Ansible to apply security updates and harden the systems.

Outcome:

- A fully automated infrastructure with multiple VMs on VMware, deploying a multi-tier application.
- Centralized monitoring and logging for system health and performance.

- Secure and efficient infrastructure with automated backups and firewall configurations.

Project Structure



Define the Inventory, Setup and Provisioning VMs

mohamed@master:~/lab2/inventory — vim hosts

```
[web]
192.168.250.128
[db]
192.168.250.141
```

"hosts" 5L, 44B 5,0-1 All

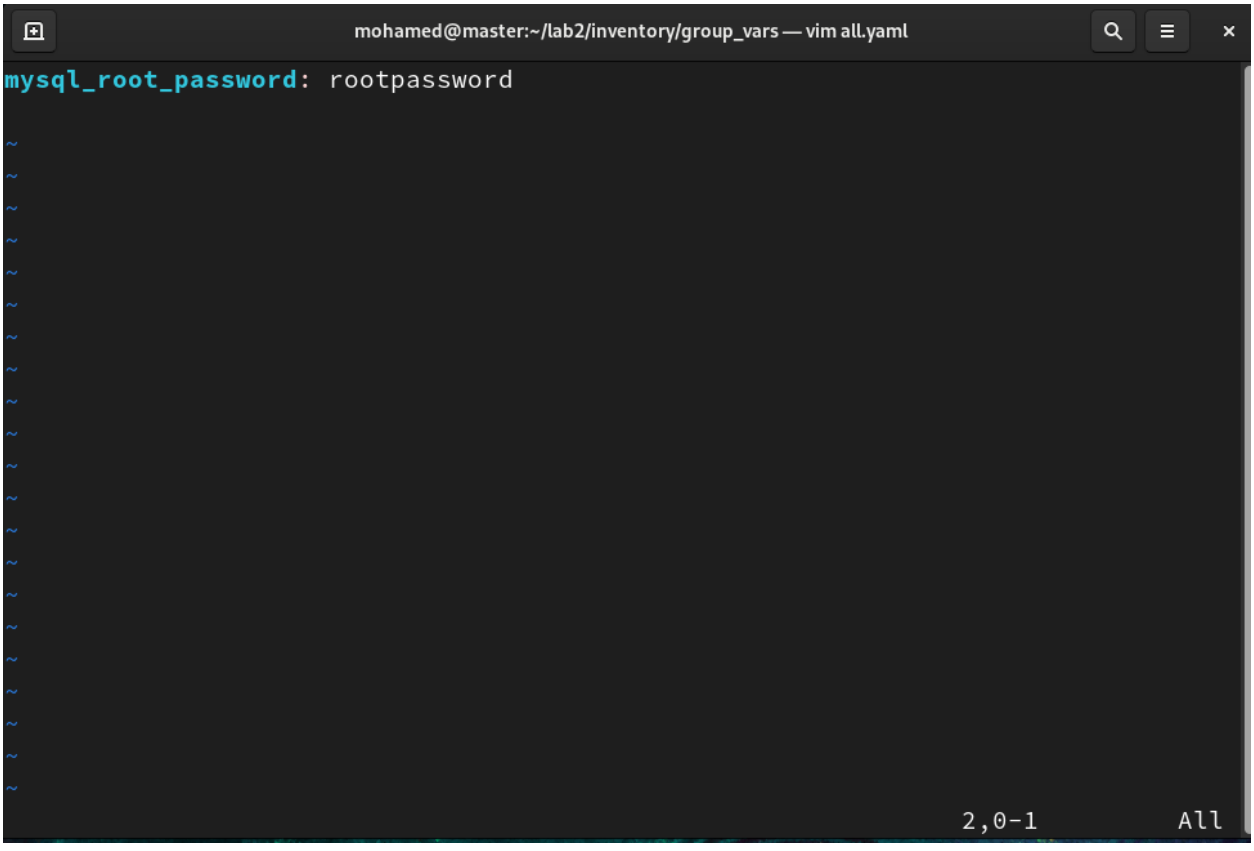
mohamed@master:~/lab2 — vim ansible.cfg

```
[defaults]
inventory = /home/mohamed/lab2/inventory
remote_user = devops

[privilege_escalation]
become = True
become_method = sudo
become_user = root
become_ask_pass = False
```

"ansible.cfg" 10L, 176B 1,1 All

```
[mohamed@master lab2]$ ansible all -m ping
192.168.250.141 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
192.168.250.128 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

Set Global Variables and use Ansible Vault



```
mohamed@master:~/Ansible_Final_Project/group_vars — vim all.yml
$ANSIBLE_VAULT;1.1;AES256
36363136393836356463623966383031316666633338626334646332313966326664303637313465
66383832663734353631313830643862613265386562326 10a3865656164356262373238366330 32
38383663663561376464346163393833383737326262343638313435366662396463373433343662
343938656662643764 0a383564363838333730646136623965356461663966643665353134383636
33353632306262663434616565363738323235636664336464646530323666663735616630623063
39386366613036613037313466346438616436346665626563 38
~
~
~
~
~
~
~
~
~
~
~
~
```

Create Roles

Apache Role (Web Server)

```
mohamed@master:~/lab2/roles/apache/tasks — vim main.yaml
---
- name: Install Apache
  yum:
    name: httpd
    state: present

- name: Install PHP
  yum:
    name: php
    state: present

- name: Copy index.php template
  template:
    src: index.php.j2
    dest: /var/www/html/index.php
    mode: 0644

- name: Start and enable Apache
  systemd:
    name: httpd
    state: started
    enabled: yes

:wq
```

PHP application template

mohamed@master:~/lab2/roles/apache/templates — vim index.php.j2

```
<?php
$conn = new mysqli('node2', 'root', '{{ mysql_root_password }}');
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully to MySQL!";
?>
```

8,0-1 All

Apache Configuration

mohamed@master:~/lab2/roles/apache/files — vim main.yml

```
- name: Copy custom Apache configuration file
  copy:
    src: httpd.conf
    dest: /etc/httpd/conf/httpd.conf
    owner: root
    group: root
    mode: 0644

- name: Restart Apache to apply new configuration
  systemd:
    name: httpd
    state: restarted
```

-- INSERT -- 4,1 All

MySQL Role (Database Server)

mohamed@master:~/lab2/roles/mysql/tasks — vim main.yml

```
---
- name: Install Python 3 and PyMySQL
  yum:
    name:
      - python3
      - python3-PyMySQL
    state: present

- name: Install MySQL
  yum:
    name: mysql-server
    state: present

- name: Start and enable MySQL
  systemd:
    name: mysqld
    state: started
    enabled: yes

- name: Set MySQL root password
  mysql_user:
    name: root
    password: "{{ mysql_root_password }}"
    login_unix_socket: /var/lib/mysql/mysql.sock
    state: present

~
```

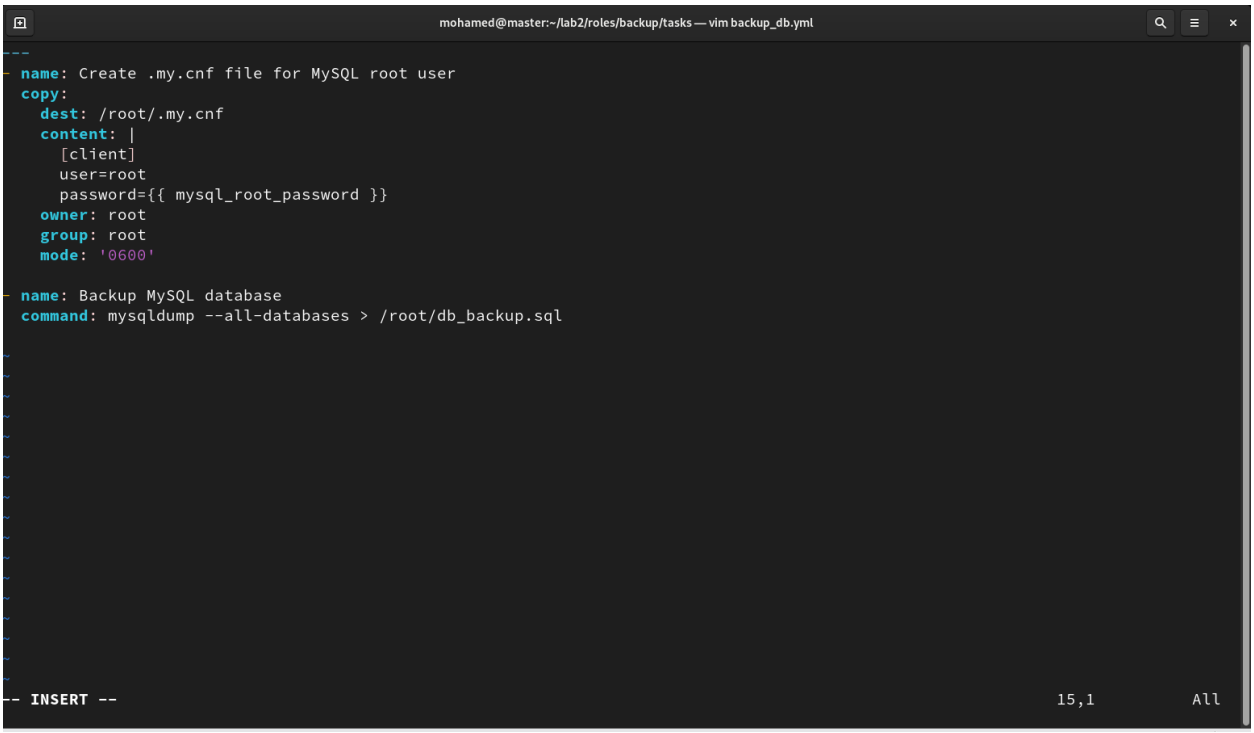
Firewall Role (For Both Nodes)

mohamed@master:~/lab2/roles/firewall/tasks — vim main.yml

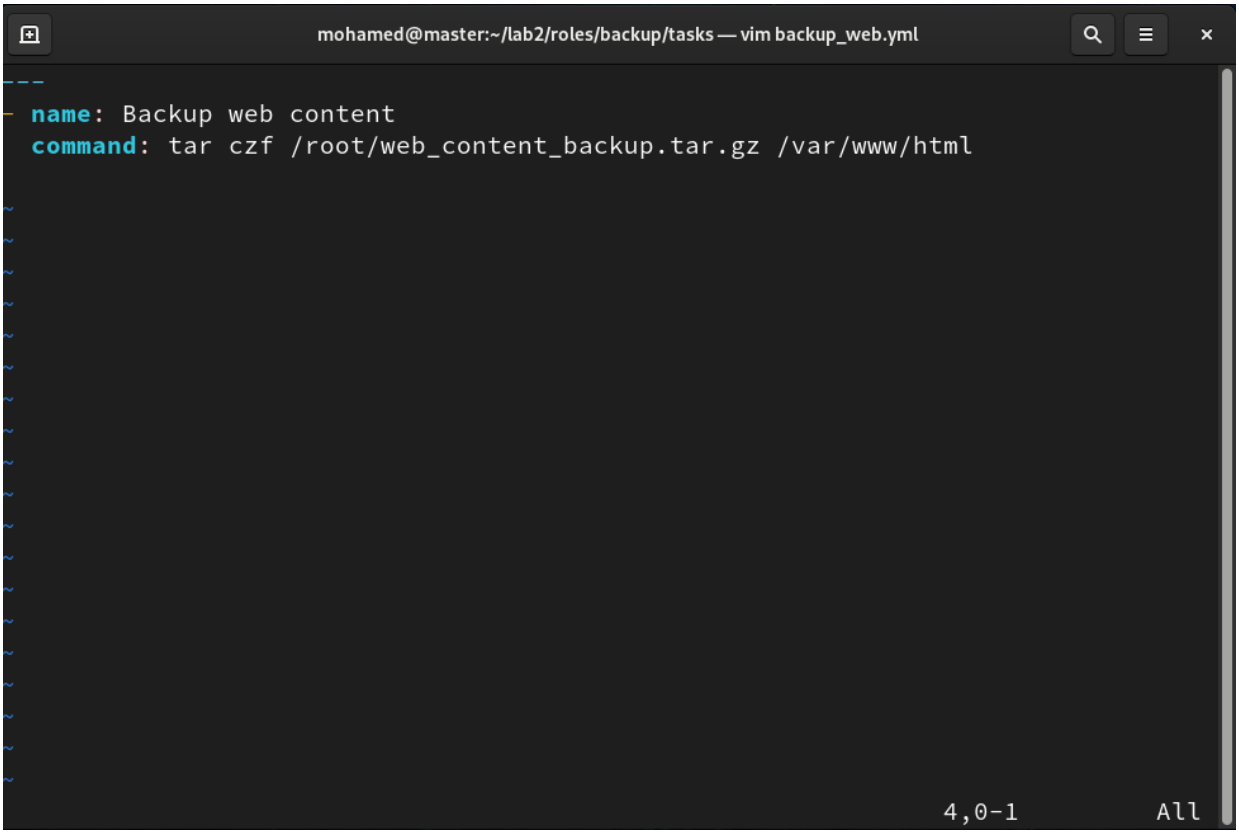
```
---
- name: Open HTTP port for Apache
  firewalld:
    port: 80/tcp
    permanent: yes
    state: enabled

- name: Open MySQL port
  firewalld:
    port: 3306/tcp
    permanent: yes
    state: enabled

- name: Reload firewall
  firewalld:
    state: reloaded

~
~
~
~
~
~
                                                    17,0-1        All
```

Backup Role

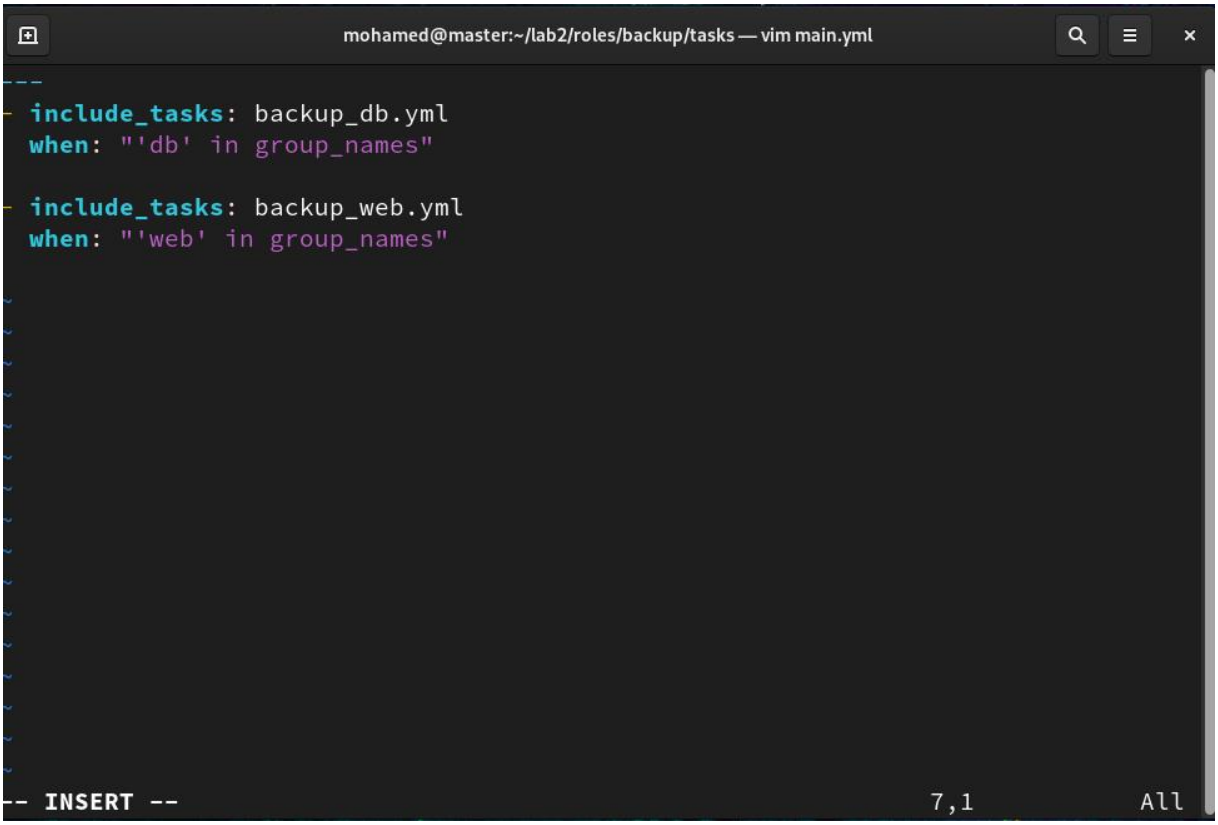
- **For MySQL Database**



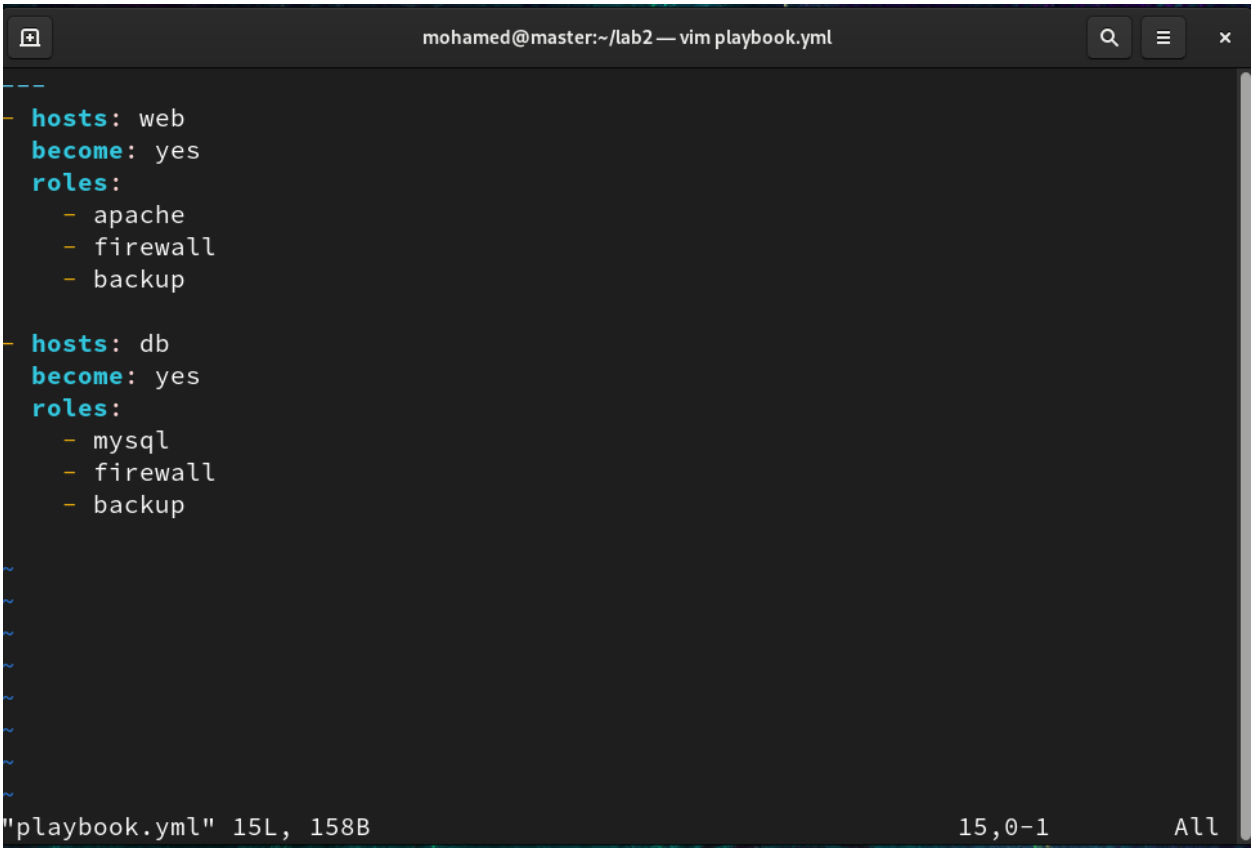
- **For Apache Web Content**



- **Backup Main Role**



Create the Playbook



Use Ansible Vault

```
[mohamed@master lab2]$ ansible-vault encrypt group_vars/all.yml
New Vault password:
Confirm New Vault password:
Encryption successful
```

Apply The Playbook

```
[mohamed@master lab2]$ ansible-playbook -i inventory/hosts playbook.yml --ask-vault-pass
Vault password:

PLAY [web] *********************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [192.168.250.128]

TASK [apache : Install Apache] *************************************************************************************************
ok: [192.168.250.128]

TASK [apache : Install PHP] ****************************************************************************************************
ok: [192.168.250.128]

TASK [apache : Copy index.php template] ****************************************************************************************
ok: [192.168.250.128]

TASK [apache : Start and enable Apache] ****************************************************************************************
ok: [192.168.250.128]

TASK [firewall : Open HTTP port for Apache] ************************************************************************************
ok: [192.168.250.128]

TASK [firewall : Open MySQL port] **********************************************************************************************
ok: [192.168.250.128]

TASK [firewall : Reload firewalld service] *************************************************************************************
changed: [192.168.250.128]

TASK [backup : include_tasks] **************************************************************************************************
skipping: [192.168.250.128]

TASK [backup : include_tasks] **************************************************************************************************
included: /home/mohamed/lab2/roles/backup/tasks/backup_web.yml for 192.168.250.128

TASK [backup : Backup web content] *********************************************************************************************
changed: [192.168.250.128]
```

```
TASK [backup : Backup web content] *********************************************************************************************
changed: [192.168.250.128]

PLAY [db] **********************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
ok: [192.168.250.141]

TASK [mysql : Install Python 3 and PyMySQL] ************************************************************************************
ok: [192.168.250.141]

TASK [mysql : Install MySQL] ***************************************************************************************************
ok: [192.168.250.141]

TASK [mysql : Start and enable MySQL] ******************************************************************************************
ok: [192.168.250.141]

TASK [mysql : Set MySQL root password] *****************************************************************************************
ok: [192.168.250.141]

TASK [firewall : Open HTTP port for Apache] ************************************************************************************
ok: [192.168.250.141]

TASK [firewall : Open MySQL port] **********************************************************************************************
ok: [192.168.250.141]

TASK [firewall : Reload firewalld service] *************************************************************************************
changed: [192.168.250.141]

TASK [backup : include_tasks] **************************************************************************************************
included: /home/mohamed/lab2/roles/backup/tasks/backup_db.yml for 192.168.250.141

TASK [backup : Create .my.cnf file for MySQL root user] ************************************************************************
ok: [192.168.250.141]

TASK [backup : Backup MySQL database] ******************************************************************************************
changed: [192.168.250.141]

TASK [backup : include_tasks] **************************************************************************************************
skipping: [192.168.250.141]

PLAY RECAP *********************************************************************************************************************
192.168.250.128            : ok=10   changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
192.168.250.141            : ok=11   changed=2    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```