Aim: Configuration Management with Ansible.

Objectives:

- Install and configure Ansible on Ubuntu.
- Set up SSH key-based authentication.
- Configure Ansible inventory.
- Run an Ansible playbook to manage configurations.

Tools Used: Virtual box, Ubuntu , Ansible

Concepts:

- Configuration Management: The process of maintaining and controlling software and system configurations consistently.
- Ansible Automation: A tool that automates IT tasks like configuration management, application deployment using YAML playbooks.
- SSH Key-Based Authentication: A secure method of logging into remote systems without passwords using cryptographic key pairs.

Problem Statement

To Install and configure Ansible on Ubuntu. To Set up SSH key-based authentication. To Configure Ansible inventory. To Run an Ansible playbook to manage configurations.

Process:

1) To configure the PPA on your system and install Ansible run these commands:

```
$ sudo apt update
$ sudo apt install software-properties-common
$ sudo add-apt-repository --yes --update ppa:ansible/ansible
$ sudo apt install ansible
```

And then check the ansible version

```
atharva@atharva-VirtualBox:~$ ansible --version
ansible [core 2.17.9]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/atharva/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/atharva/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
```

2) Now install SSH

   sudo apt update
   sudo apt install openssh-server -y
   sudo systemctl enable ssh
   sudo systemctl start ssh

3) Set Up SSH Key-Based Authentication
   ssh-keygen

4) Copy the SSH Key to Client:

```
atharva@atharva-VirtualBox:~/ansible-project$ ssh-copy-id -i ~/.ssh/id_rsa.pub atharva@atharva-VirtualBox
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/atharva/.ssh/id_rsa.pub"
The authenticity of host 'atharva-virtualbox (127.0.1.1)' can't be established.
ED25519 key fingerprint is SHA256:/n3o6y4g1f83irFQo+nImgPtmqwpKGMHwEyLGLDYoBU.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
sign_and_send_pubkey: signing failed for RSA "/home/atharva/.ssh/id_rsa" from agent: agent refused operation
atharva@atharva-virtualbox's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'atharva@atharva-VirtualBox'"
and check to make sure that only the key(s) you wanted were added.
```

5) Configuring Ansible on the Server
   use:
   - sudo nano /etc/ansible/hosts

   <u>Now add:</u>

   [webserver]
   server1 ansible_host=127.0.0.1 ansible_user=atharva ansible_ssh_private_key_file=~/.ssh/id_rsa

   127.0.0.1 = IP of the machine
   atharva = machine name

~/.ssh/id_rsa = path to the private key

Also Add:
[servers]
server1 ansible_host=127.0.0.1

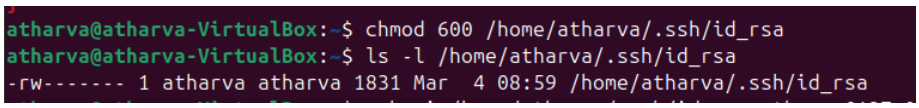
[all:vars]
ansible_ssh_private_key_file=/home/atharva/.ssh/id_rsa

Now save the file

6) Now set permission to the private key
Only you (the owner) can read and write the private SSH key.
No other users (not even those in your group) can read, write, or execute the file.
This is essential for security because SSH private keys must be kept secret to prevent unauthorized access to serve

Use command:
chmod 600 /home/atharva/.ssh/id_rsa

This will result in:
atharva@atharva-VirtualBox:~$ ls -l /home/atharva/.ssh/id_rsa
-rw------- 1 atharva atharva 1831 Mar 4 08:59 /home/atharva/.ssh/id_rsa



7) connect your own machine (localhost) via SSH

Use command:
ssh -i /home/atharva/.ssh/id_rsa atharva@127.0.0.1

```
atharva@atharva-VirtualBox:~$ ssh -i /home/atharva/.ssh/id_rsa atharva@127.0.0.1
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.11.0-19-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

35 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Mar  7 12:53:33 2025 from 127.0.0.1
```

8) Now ping the server
   Use command:
   ansible webserver -m ping

```
atharva@atharva-VirtualBox:~$ ansible webserver -m ping
[WARNING]: Platform linux on host server1 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of
another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
server1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
```

9) Now create a playbook in a directory
   Use command:
   - mkdir ansible-project →to create the directory
   - nano package-playbook.yaml →to create the playbook
     In the playbook add the below code and save.

```yaml
---
- name: Install Apache on Managed Node
  hosts: webserver
  become: yes
  tasks:
    - name: Install Apache2
      apt:
        name: apache2
        state: present
```

10) Now run the playbook
Use command:
ansible-playbook package-playbook.yaml --ask-become-pass

```
atharva@atharva-VirtualBox:~/ansible-project$ ansible-playbook package-playbook.yaml --ask-become-pass
BECOME password:

PLAY [Install Apache on Managed Node] ******************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************
[WARNING]: Platform linux on host server1 is using the discovered Python interpreter at /usr/bin/python3.12, but future installation of
another Python interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-
core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [server1]

TASK [Install Apache2] *********************************************************************************************************
changed: [server1]

PLAY RECAP *********************************************************************************************************************
server1                    : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Conclusion: This setup automates configuration management using Ansible, making deployments more efficient and reducing manual effort. By using SSH key-based authentication and playbooks, Ansible enables seamless server management, ensuring consistency across environments.