

What is Ansible?

Ansible is an **open-source automation tool** used for:

- **Configuration management** (installing software, editing config files),
- **Application deployment**,
- **Server provisioning**, and
- **IT orchestration** (coordinating multiple systems).

It helps automate repetitive tasks across **many servers** using **simple YAML scripts** (called playbooks).

Ansible doesn't require installing agents on target systems—it uses **SSH** to communicate.

Real-Time Example Use Case

Example Scenario: Deploy Apache Web Server to Multiple Servers

Imagine you are a **DevOps Engineer** managing 100 Linux servers. You want to install and start the **Apache2 web server** on all of them.

Instead of connecting to each server manually and running:

```
sudo apt update
```

```
sudo apt install apache2 -y
```

```
sudo systemctl start apache2
```

You just write one **Ansible playbook**, and Ansible will run all commands automatically on all servers.

Step 1: Update System Packages

```
sudo apt update
```

```
Hit:1 http://archive.ubuntu.com/ubuntu focal InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Step 2: Install Software Properties Common

```
sudo apt install software-properties-common -y
```

Step 3: Add Ansible PPA Repository

```
sudo add-apt-repository --yes --update ppa:ansible/ansible
```

Step 4:

Install Ansible

```
sudo apt install ansible -y
```

Verify Installation

```
ansible --version
```

```
ansible [core 2.15.3]
  python version = 3.10.12
  jinja version = 3.1.2
```

Create a Playbook to Install Apache2

Step 1: Create a folder for the project

```
mkdir apache2
```

```
cd apache2
```

Step 2: Create a Playbook File

```
nano apache2-install.yml
```

Then paste the content

```
---
```

```
- name: Install and Start Apache2 on Localhost
```

```
  hosts: localhost
```

```
  become: true
```

```
  connection: local
```

```
  tasks:
```

```
    - name: Update APT package index
```

```
      apt:
```

```
        update_cache: yes
```

```
    - name: Install Apache2 web server
```

```
      apt:
```

name: apache2

state: present

- name: Ensure Apache2 service is running and enabled

service:

name: apache2

state: started

enabled: yes

- name: Display Apache service status

command: systemctl status apache2

register: apache_status

ignore_errors: yes

Create Inventory File

nano hosts

Add this line

[local]

localhost ansible_connection=local

Verify inventory:

ansible all -i hosts -m ping

Output

```
localhost | SUCCESS => {  
    "changed": false,  
    "ping": "pong"  
}
```

Run the Playbook

Check syntax before running:

```
ansible-playbook -i hosts apache2-install.yml --syntax-check
```

Output

```
playbook: apache2-install.yml
```

Run the playbook:

```
ansible-playbook -i hosts apache2-install.yml
```

Expected Output

```
PLAY [Install and Start Apache2 on Localhost] *****

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

TASK [Update APT package index] *****
changed: [localhost]

TASK [Install Apache2 web server] *****
changed: [localhost]

TASK [Ensure Apache2 service is running and enabled] *****
ok: [localhost]

TASK [Display Apache service status] *****
ok: [localhost]

PLAY RECAP *****
localhost                : ok=5    changed=2    unreachable=0    failed=0
```

Verify Apache Installation

```
sudo systemctl status apache2
```

```
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled)
   Active: active (running)
   Main PID: 1576 (apache2)
```

Now go to browser

<http://localhost>

You'll see the **Apache2 Ubuntu Default Page** — proof that Ansible successfully installed and started it.