Lesson End Project

Simplifying Inventory with Ranges

Project Agenda: To perform simplification of inventory using ranges to define multiple hosts in a more concise and manageable way

Description: As a DevOps engineer at a tech company, you manage a large fleet of servers. In your current project, you maintain a cluster of 100 web servers which need frequent updates. Using range patterns in Ansible inventory files, you will simplify and streamline inventory management. This reduces redundancy, makes files easier to manage, and ensures efficient, consistent updates across all servers with minimal effort.

Tools required: Ansible

Prerequisites: None

Expected Deliverables: A simplified Ansible inventory file using ranges to define multiple hosts, a working Ansible playbook that runs tasks against the defined inventory groups, and successful execution and verification of the playbook with appropriate logs.

Steps to be followed:

1. Install Ansible on the server
2. Create an inventory file
3. Test the inventory
4. Create a simple playbook
5. Run the playbook

Step 1: Install Ansible on the server

1.1 Use the following commands to install Ansible:

sudo apt update

sudo apt install ansible -y

```
poojahksimplile@ip-172-31-36-118:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 https://download.docker.com/linux/ubuntu jammy InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:6 https://pkg.jenkins.io/debian-stable binary/ Release
Get:8 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:7 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb  InRelease
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1794 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.7 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [604 B]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1101 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [25.4 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [444 B]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1587 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.0 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [572 B]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [884 kB]
Get:20 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [18.9 kB]
Get:21 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [228 B]
Fetched 5700 kB in 2s (3746 kB/s)
```

```
poojahksimplile@ip-172-31-36-118:~$ sudo apt install ansible -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ansible is already the newest version (5.10.0-1ppa~focal).
0 upgraded, 0 newly installed, 0 to remove and 79 not upgraded.
```

Step 2: Create an inventory file

2.1 Use the following command to navigate to your directory inventory file:

cd /home/Username/.ssh

```
poojahksimplile@ip-172-31-36-118:~$ cd /home/              /.ssh
poojahksimplile@ip-172-31-36-118:~/.ssh$
```

Note: Make sure to replace **Username** with the actual username.

2.2 Use the below command to create an inventory file:

nano host

```
poojahksimplile@ip-172-31-36-118:~/.ssh$ nano hosts
poojahksimplile@ip-172-31-36-118:~/.ssh$
```

2.3 When the file opens, add the below data, save, and exit the file:

 # /home/username/.ssh/hosts

[webservers]

192.168.1.10

192.168.1.11

192.168.1.12

192.168.1.13

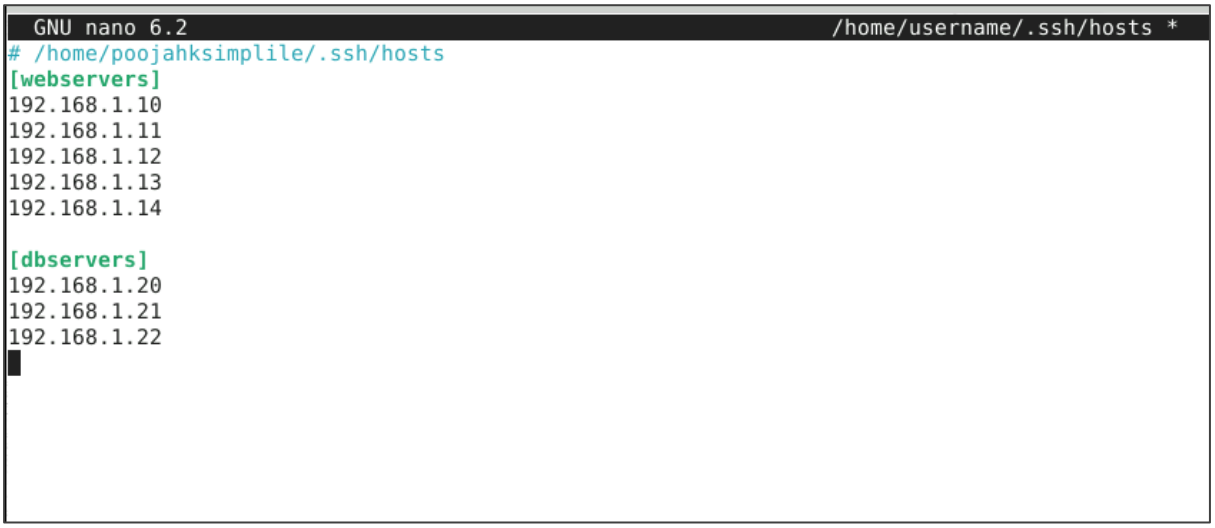
192.168.1.14

[dbservers]

192.168.1.20

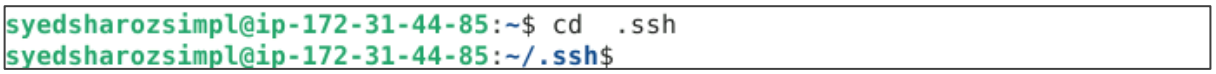
192.168.1.21

192.168.1.22



2.4 Run the following command to edit the **/etc/hosts** file:

sudo nano /etc/hosts



2.5 When the file opens, add the below data, save it, and exit the file server:

192.168.1.10 web01.example.com

192.168.1.11 web02.example.com

192.168.1.12 web03.example.com

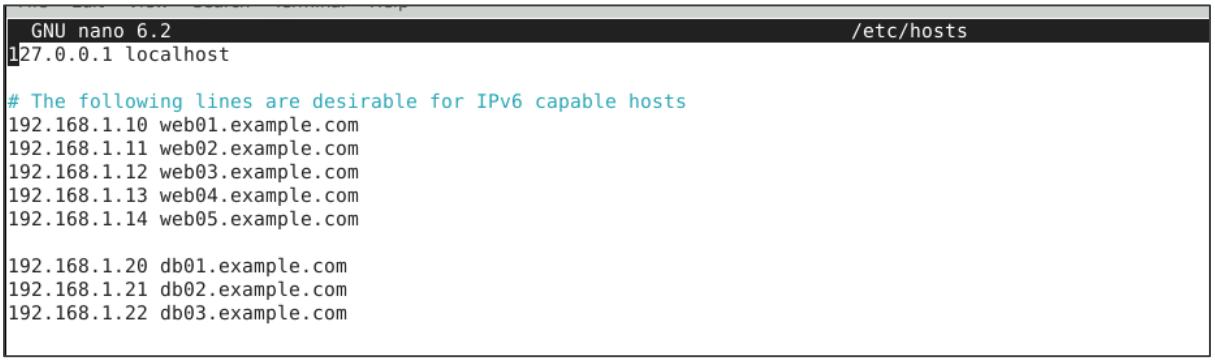
192.168.1.13 web04.example.com

192.168.1.14 web05.example.com

192.168.1.20 db01.example.com

192.168.1.21 db02.example.com

192.168.1.22 db03.example.com



Step 3: Test the inventory

3.1 Use the following command to test the inventory:

ansible-inventory -i hosts --list

```
poojahksimplile@ip-172-31-36-118:~/.ssh$ ansible-inventory -i hosts --list
{
    "_meta": {
        "hostvars": {}
    },
    "all": {
        "children": [
            "dbservers",
            "ungrouped",
            "webservers"
        ]
    },
    "dbservers": {
        "hosts": [
            "192.168.1.20",
            "192.168.1.21",
            "192.168.1.22"
        ]
    },
    "webservers": {
        "hosts": [
            "192.168.1.10",
            "192.168.1.11",
            "192.168.1.12",
            "192.168.1.13",
            "192.168.1.14"
        ]
    }
}
```

Step 4: Create a simple playbook

4.1 Run the following command to create a playbook:

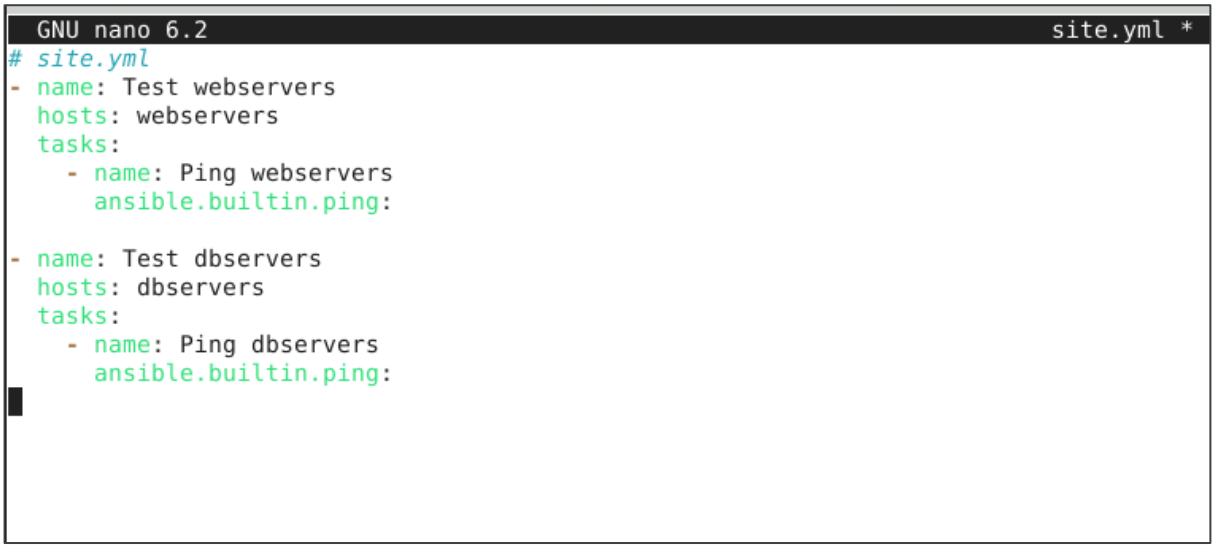
nano site.yml

```
poojahksimplile@ip-172-31-36-118:~/.ssh$ nano site.yml
```

4.2 Enter the following details in the playbook:

```
# site.yml
- name: Test webservers
  hosts: webservers
  tasks:
    - name: Ping webservers
      ansible.builtin.ping:

- name: Test dbservers
  hosts: dbservers
  tasks:
    - name: Ping dbservers
      ansible.builtin.ping:
```



Step 5: Run the playbook

5.1 Run the playbook by executing the below command:
ansible-playbook -i hosts site.yml

```
poojahksimplile@ip-172-31-36-118:~/.ssh$ ansible-playbook -i hosts site.yml

PLAY [Test webservers] *********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
```

By following these steps, you have successfully simplified inventory with ranges for defining multiple hosts, making it more concise and manageable, especially when dealing with many hosts.