System Provisioning and Configuration Management Lab

Submitted to:

Dr. Hitesh Kumar Sharma

Submitted by:
Saumya Sehgal
500105655
R2142220635
Devops B2(NH)

EXPERIMENT 3

Lab Exercise: Creating Static Host Inventory

Objective: To demonstrate ad hoc commands for quickly executing tasks on remote servers without writing full playbooks

Tools required: Ansible, Ubuntu OS

Prerequisites: None

Steps to be followed:

1. Generate SSH key pair on the main node
2. Copy the SSH key on the other two nodes
3. Update the host file with the host IP address
4. Establish connectivity between specified hosts and the Ansible server
5. Gather System Information Using Ad-Hoc Commands

Step 1: Establish connectivity between specified hosts and the Ansible server

1.1 Run the following command to verify connectivity to all servers listed under the **webservers** group in your Ansible hosts file: **ansible -m ping dbservers**

```
ubuntu@ip-172-31-15-82:~$ ansible -m ping dbbservers
localhost | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
65.0.183.70 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
172.31.15.82 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

1.2 Use the following command to check the number of hosts in the host file:
ansible all --list-hosts

```
ubuntu@ip-172-31-15-82:~$ ansible all --list-hosts
  hosts (3):
    localhost
    65.0.183.70
    172.31.15.82
```

Step 2: Gather System Information Using Ad Hoc Commands

2.1 Run the following command to obtain the uptime from all managed hosts using an ad hoc command: **ansible all -m shell -a uptime**

```
ubuntu@ip-172-31-15-82:~$ ansible all -m shell -a uptime
172.31.15.82 | CHANGED | rc=0 >>
 01:01:49 up 52 min,  6 users,  load average: 0.02, 0.03, 0.01
65.0.183.70 | CHANGED | rc=0 >>
 01:01:49 up 52 min,  6 users,  load average: 0.02, 0.03, 0.01
localhost | CHANGED | rc=0 >>
 01:01:49 up 52 min,  6 users,  load average: 0.02, 0.03, 0.01
```

2.2 Similarly, execute the below command to obtain detailed information about memory usage on all hosts: **ansible all -m shell -a "free -m"**

```
ubuntu@ip-172-31-15-82:~$ ansible all -m shell -a "free -m"
172.31.15.82 | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:            957         495          82           1         562         461
Swap:             0           0           0
localhost | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:            957         496          81           1         562         460
Swap:             0           0           0
65.0.183.70 | CHANGED | rc=0 >>
              total        used        free      shared  buff/cache   available
Mem:            957         496          81           1         562         460
Swap:             0           0           0
ubuntu@ip-172-31-15-82:~$
```

You will see that Ansible logs in to each machine in turn and runs the uptime command, returning the current uptime output.

By following these steps, you have successfully demonstrated how to use ad hoc commands for quickly executing tasks on remote servers without the need for full playbooks.