

Lesson 03 Demo 02

Executing Ansible Modules

Objective: To demonstrate the use of the Ansible module for server configuration and management

Tools required: Linux terminal

Prerequisites: None

Steps to be followed:

1. Use Ansible modules for server configuration

Step 1: Use Ansible modules for server configuration

- 1.1 Run the following command using the **setup** module to collect and display detailed system information about the servers:

ansible -m setup webserver

```
poojahksimplile@ip-172-31-79-37:~$ ansible -m setup webserver
localhost | SUCCESS => {
  "ansible_facts": {
    "ansible_all_ipv4_addresses": [
      "172.31.79.37",
      "172.17.0.1"
    ],
    "ansible_all_ipv6_addresses": [
      "fe80::14ff:ea3f:fef3:28bf"
    ],
    "ansible_apparmor": {
      "status": "enabled"
    },
    "ansible_architecture": "x86_64",
    "ansible_bios_date": "10/16/2017",
    "ansible_bios_vendor": "Amazon EC2",
    "ansible_bios_version": "1.0",
    "ansible_board_asset_tag": "i-086276f0d4ecf97a3",
    "ansible_board_name": "NA",
    "ansible_board_serial": "NA",
    "ansible_board_vendor": "Amazon EC2",
    "ansible_board_version": "NA",
    "ansible_chassis_asset_tag": "Amazon EC2",
    "ansible_chassis_serial": "NA",
    "ansible_chassis_vendor": "Amazon EC2",
    "ansible_chassis_version": "NA",
    "ansible_cmdline": {
      "BOOT_IMAGE": "/boot/vmlinuz-6.5.0-1016-aws",
      "console": "ttyS0",
```

1.2 Run the following command using the **shell** module to retrieve server hostnames:

ansible webserver -m shell -a 'hostname'

```
poojahksimplile@ip-172-31-79-37:~$ ansible webserver -m shell -a 'hostname'
localhost | CHANGED | rc=0 >>
ip-172-31-79-37
poojahksimplile@ip-172-31-79-37:~$
```

1.3 Run the following command using the **apt** module to install Git on servers:

ansible webserver -m apt -a 'name=git state=present' --become

```
poojahksimplile@ip-172-31-79-37:~$ ansible webserver -m apt -a 'name=git state=present' --become
localhost | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "cache_update_time": 1714740694,
  "cache_updated": false,
  "changed": false
}
poojahksimplile@ip-172-31-79-37:~$
```

1.4 Run the following command using the **file** module to create a secured root file:

ansible webserver -m file -a 'dest=/root/sample.txt state=touch mode=600 owner=root group=root' --become

```
poojahksimplile@ip-172-31-79-37:~$ ansible webserver -m file -a 'dest=/root/sample.txt state=touch mode=600 owner=root group=root' --become
localhost | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": true,
  "dest": "/root/sample.txt",
  "gid": 0,
  "group": "root",
  "mode": "0600",
  "owner": "root",
  "size": 0,
  "state": "file",
  "uid": 0
}
poojahksimplile@ip-172-31-79-37:~$
```

By following these steps, you have successfully managed server configurations using Ansible modules.