

QUESTION 5

5. Ansible

[< Previous](#)[Next >](#)

Create a simple playbook that updates all packages on a remote host as a root user.

SOLUTION:

1. Created hosts file to define groups of remote machines and their connection details for Ansible to manage.

```
kaushal@master-vm:~$ cd ansible-update-packages/
kaushal@master-vm:~/ansible-update-packages$ ls
hosts  update_packages.yml
kaushal@master-vm:~/ansible-update-packages$ cat hosts
[all]
host1 ansible_host=192.168.153.131
host2 ansible_host=192.168.153.133
```

2. Yaml file for updating packing on machines

```
kaushal@master-vm:~/ansible-update-packages$ cat update_packages.yml
---
- name: Update packages on remote hosts
  hosts: all
  become: true
  tasks:
    - name: Update all packages (Debian/Ubuntu)
      apt:
        update_cache: yes
        upgrade: dist
        when: ansible_os_family == "Debian"
    - name: Update all packages (RedHat/CentOS/Amazon)
      yum:
        name: "*"
        state: latest
        when: ansible_os_family == "RedHat"
```

3. Output for the SUCCESSFUL UPDATE PACKAGES

```
kaushal@master-vm:~/ansible-update-packages$ ansible-playbook -i hosts update_packages.yml --ask-pass --ask-become-pass
SSH password:
BECOME password[defaults to SSH password]:
PLAY [Update packages on remote hosts] *****
TASK [Gathering Facts] *****
ok: [host2]
ok: [host1]
TASK [Update all packages (Debian/Ubuntu)] *****
changed: [host1]
changed: [host2]
TASK [Update all packages (RedHat/CentOS/Amazon)] *****
skipping: [host1]
skipping: [host2]
PLAY RECAP *****
host1      : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
host2      : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
kaushal@master-vm:~/ansible-update-packages$
```

FILES USED:

1. HOST FILE

```
kaushal@master-vm:~/ansible-update-packages$ cat hosts
[all]
host1 ansible_host=192.168.153.131
host2 ansible_host=192.168.153.133
```

2. UPDATE.PACKAGE.YAML FILE FOR UPDATEING THE PACKAGE

```
---
- name: Update packages on remote hosts
  hosts: all
  become: true
  tasks:
    - name: Update all packages (Debian/Ubuntu)
      apt:
        update_cache: yes
        upgrade: dist
        when: ansible_os_family == "Debian"

    - name: Update all packages (RedHat/CentOS/Amazon)
      yum:
        name: "*"
        state: latest
        when: ansible_os_family == "RedHat"
```