

DevOps Quest: One Week Deployment Challenge :Day 03

Step1:

add host entry in /etc/ansible/hosts

```
ubuntu@master-node:~$ cd /etc/ansible/
ubuntu@master-node:/etc/ansible$ ls
ansible.cfg  hosts  roles
ubuntu@master-node:/etc/ansible$ cat hosts
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups
#
# Ex 1: Ungrouped hosts, specify before any group headers:
## green.example.com
## blue.example.com
# 192.168.100.1
## 192.168.100.10
#
# Ex 2: A collection of hosts belonging to the 'webservers' group:
[webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
192.168.122.208 ansible_user=satishvm2 ansible_ssh_pass=root
#
# If you have multiple hosts following a pattern, you can specify
# them like this:
## www[001:006].example.com
#
# You can also use ranges for multiple hosts:
## db-[99:101]-node.example.com
#
# Ex 3: A collection of database servers in the 'dbservers' group:
```

```
[webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
192.168.122.208 ansible_user=satishvm2 ansible_ssh_pass=root
```

Step2:

create docker ansible file

```
root@master-node:/home/ubuntu/devops/ansible# cat dockerinstall.yml
---
- name: Install Docker on Ubuntu
  hosts: webserver
  become: yes
  become_method: sudo
  become_user: root
  vars_prompt:
    - name: ansible_become_pass
      prompt: "Enter sudo password"
      private: yes
  tasks:
    - name: Update apt package index
      apt:
        update_cache: yes

    - name: Install required packages
      apt:
        name: "{{ item }}"
        state: present
      loop:
        - apt-transport-https
        - ca-certificates
        - curl
        - gnupg-agent
        - software-properties-common

    - name: Add Docker GPG key
      apt_key:
        url: https://download.docker.com/linux/ubuntu/gpg

    - name: Add Docker repository
      apt_repository:
        repo: deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable

    - name: Update apt package index
      apt:
        update_cache: yes

    - name: Install Docker
      apt:
        name: docker-ce
        state: present

    - name: Add user to Docker group
      user:
        name: "{{ ansible_user }}"
        groups: docker
        append: yes
        when: ansible_os_family == 'Debian'
```

root@master-node:/home/ubuntu/devops/ansible# █

/

vi dockerinstall.yml

ansible Playbook

```
- name: Install Docker on Ubuntu
  hosts: webservers
  become: yes
  become_method: sudo
  become_user: root
  vars_prompt:
    - name: ansible_become_pass
      prompt: "Enter sudo password"
      private: yes
  tasks:
    - name: Update apt package index
      apt:
        update_cache: yes

    - name: Install required packages
      apt:
        name: "{{ item }}"
        state: present
      loop:
        - apt-transport-https
        - ca-certificates
        - curl
        - gnupg-agent
        - software-properties-common

    - name: Add Docker GPG key
      apt_key:
        url: https://download.docker.com/linux/ubuntu/gpg

    - name: Add Docker repository
      apt_repository:
        repo: deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable

    - name: Update apt package index
      apt:
        update_cache: yes

    - name: Install Docker
      apt:
        name: docker-ce
        state: present

    - name: Add user to Docker group
      user:
        name: "{{ ansible_user }}"
        groups: docker
        append: yes
```

when: ansible_os_family == 'Debian'

Ansible Command

ansible-playbook -i /etc/ansible/hosts dockerinstall.yml

```
root@master-node:/home/ubuntu/devops/ansible# vi dockerinstall.yml
root@master-node:/home/ubuntu/devops/ansible# ansible-playbook -i /etc/ansible/hosts dockerinstall.yml
Enter sudo password:

PLAY [Install Docker on Ubuntu] *****

TASK [Gathering Facts] *****
ok: [192.168.122.208]

TASK [Update apt package index] *****
changed: [192.168.122.208]

TASK [Install required packages] *****
ok: [192.168.122.208] => (item=apt-transport-https)
ok: [192.168.122.208] => (item=ca-certificates)
ok: [192.168.122.208] => (item=curl)
ok: [192.168.122.208] => (item=gnupg-agent)
ok: [192.168.122.208] => (item=software-properties-common)

TASK [Add Docker GPG key] *****
ok: [192.168.122.208]

TASK [Add Docker repository] *****
ok: [192.168.122.208]

TASK [Update apt package index] *****
changed: [192.168.122.208]

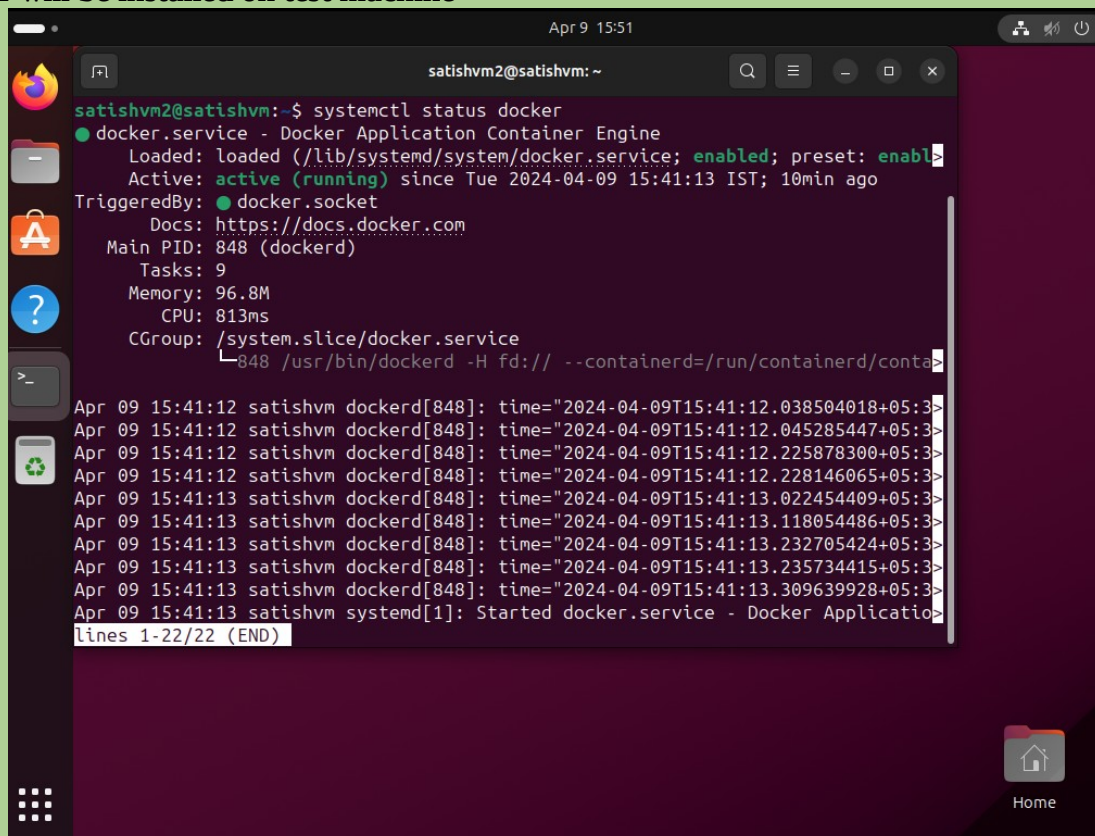
TASK [Install Docker] *****
ok: [192.168.122.208]

TASK [Add user to Docker group] *****
changed: [192.168.122.208]

PLAY RECAP *****
192.168.122.208      : ok=8    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

/

Docker will be installed on test machine



The screenshot shows a terminal window titled 'satishvm2@satishvm: ~' with a timestamp of 'Apr 9 15:51'. The user has run the command 'systemctl status docker'. The output shows that the 'docker.service' is 'loaded' and 'active (running)'. It also displays details about the service, including its main PID (848), tasks (9), memory usage (96.8M), CPU usage (813ms), and CGroup. Below this, a series of log entries are shown, indicating the successful installation and startup of Docker. The logs show the 'dockerd' process starting and the 'systemd' service starting 'docker.service'.

```
satishvm2@satishvm:~$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; preset: enabled)
   Active: active (running) since Tue 2024-04-09 15:41:13 IST; 10min ago
     TriggeredBy: ● docker.socket
        Docs: https://docs.docker.com
       Main PID: 848 (dockerd)
          Tasks: 9
         Memory: 96.8M
            CPU: 813ms
        CGroup: /system.slice/docker.service
               └─848 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/conta

Apr 09 15:41:12 satishvm dockerd[848]: time="2024-04-09T15:41:12.038504018+05:3>
Apr 09 15:41:12 satishvm dockerd[848]: time="2024-04-09T15:41:12.045285447+05:3>
Apr 09 15:41:12 satishvm dockerd[848]: time="2024-04-09T15:41:12.225878300+05:3>
Apr 09 15:41:12 satishvm dockerd[848]: time="2024-04-09T15:41:12.228146065+05:3>
Apr 09 15:41:13 satishvm dockerd[848]: time="2024-04-09T15:41:13.022454409+05:3>
Apr 09 15:41:13 satishvm dockerd[848]: time="2024-04-09T15:41:13.118054486+05:3>
Apr 09 15:41:13 satishvm dockerd[848]: time="2024-04-09T15:41:13.232705424+05:3>
Apr 09 15:41:13 satishvm dockerd[848]: time="2024-04-09T15:41:13.235734415+05:3>
Apr 09 15:41:13 satishvm dockerd[848]: time="2024-04-09T15:41:13.309639928+05:3>
Apr 09 15:41:13 satishvm systemd[1]: Started docker.service - Docker Applicatio>
lines 1-22/22 (END)
```