# Infra Optimization.

**Background of the problem statement:**

A popular payment application, EasyPay where users add money to their wallet accounts, faces an issue in its payment success rate. The timeout that occurs with the connectivity of the database has been the reason for the issue.

While troubleshooting, it is found that the database server has several downtime instances at irregular intervals. This situation compels the company to create their own infrastructure that runs in high-availability mode.

Given that online shopping experiences continue to evolve as per customer expectations, the developers are driven to make their app more reliable, fast, and secure for improving the performance of the current system.

**Requirement:**

Create a DevOps infrastructure for an e-commerce application to run on high-availability mode.

**Assigned To:** Matta Avinash Reddy ( mattavireddy@outlook.com )

Index Page

Objective 4

Application Overview: 4

1.Installation and configuration of required packages/tools on Machine

a) PIP installation 5

b) Ansible Installation 5

c) Aws-collection ansible 6

2. AWS account configuration

a) IAM User setup 7

b) Key Pair setup 7

3. Ansible configuration and scripts files

a) Create directory working space 8

b) Create ansible vault file 8

c) Create Ansible Roles 8

d) Create Ansible config file 9

e) Look at Working directory 10

4.Ansible Roles Overview

a) EC2 Role 10

b) LoadBalancer Role 11

c) Elastic_IP_Address Role 11

d) K8s_Master Role 12

e) K8s_Slave Role 13

f) Pods Roles 14

5. Source of Ansible Scripts and K8 objects files

a) Ansible Scripts GitHub Link 15

b) K8 Objects GitHub Link 15

6. Ansible Scripts Execution

a) Create a Playbook to execute all roles 15

b) EC2 role results 15

c) k8s_master Role results 16

d) LoadBalancer role results 16

e) K8s_slave role results 16

f) Pods role results (deploying k8 objects) 17

Note: Network Policy implemented between MongoDb service and API service

Horizontal Auto Scale Object is setup for UI Service

g) Elastic_Ip_Address role results 18

7. Verify Ansible Results at AWS

a) EC2 Instances 18

b) Security Group 18

c) Elastic IP Address 18

d) Load Balancer 19

8. SSH into Master Node and Verify Deployed Objects

a) SSH into Master Node 19

b) Verify K8 Version 19

c) Running Ingress services for Deployed Objects 19

d) Verify K8 Objects 19

e) Mapping Ingress Nginx Controller Port to AWS Load Balancer Instance Port 20

9. Application URL Access (http://ec2lb-822963159.ap-south-1.elb.amazonaws.com) 20

10. Creating User ‘areddy’ and Assigning permissions

a) Creating user directory under ‘/home’ folder 21

b) Generating a private key for areddy user areddy.key 21

c) Creating csr request 21

d) Certificate Signing Request file areddy_csr.yml 22

e) Miscellaneous commands for User setup 22

f) Create Role, Role Binding, Cluster Role, Cluster Role Binding for ‘areddy’ user 22

g) Verify the Role permission 24

11. ECTD Database Snapshot

a) Install ETCD 25

b) Backup ETCD database 25

c) Remove the /var/lib/etcd folder and verify pods, deployments 26

d) Restore ETCD Snapshot backup and verify pods, deployments 26

12. UI configuration setup to Horizontal Scale up on CPU memory goes beyond 30%

a) Verify HorizontalPodAutoscaler has been deployed under Pods Role 26

b) Metrics Server Setup 27

c) Install apache2-utils 27

d) Testing the Load to see the behaviour of HorizontalPodAutoScaler on UI deployment 28

Summary

a) Project and tester details 28

b) Links to the GitHub Repo 28

c) Concepts used in the Project 29

d) Unique Selling Points 29

**Objective:** An complete Automation process (Ansible tool) along with some manual intervention building AWS EC2 instances and deploying an 3 tier Application on new setup of K8 cluster with Network Policies and Auto Scaling features.

**Application:** A 3 tier application displays list of Items prices and entry form of new items.

**Database:** *MongoDb*

DockerHub Link: [Mongo - Official Image | Docker Hub]()

**API:** *.Net Core API*

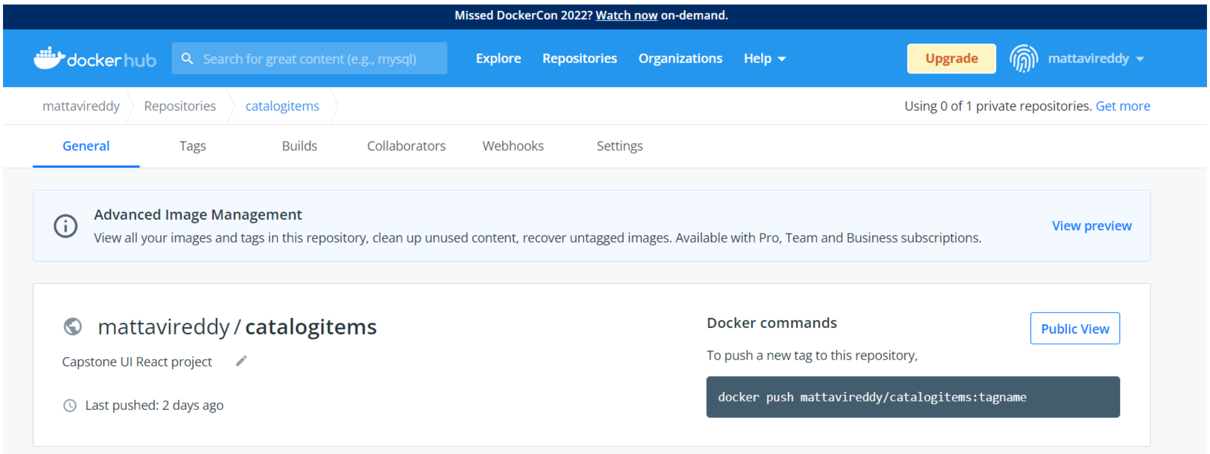
GitHub Link: [mattavicoder/catalog: Capstone project for DevOps (github.com)]()

DockerHub Link: [mattavireddy/catalog - Docker Image | Docker Hub]()

**UI:** *React-Typescript*

GitHub Link: [mattavicoder/catalog-items: Capstone Project UI (github.com)]()

DockerHub Link: [mattavireddy/catalogitems - Docker Image | Docker Hub]()



Using 0 of 1 private repositories.

Advanced Image Management

View all your images and tags in this repository, clean up unused content, recover untagged images. Available with Pro, Team and Business subscriptions.

mattavireddy / **catalogitems**

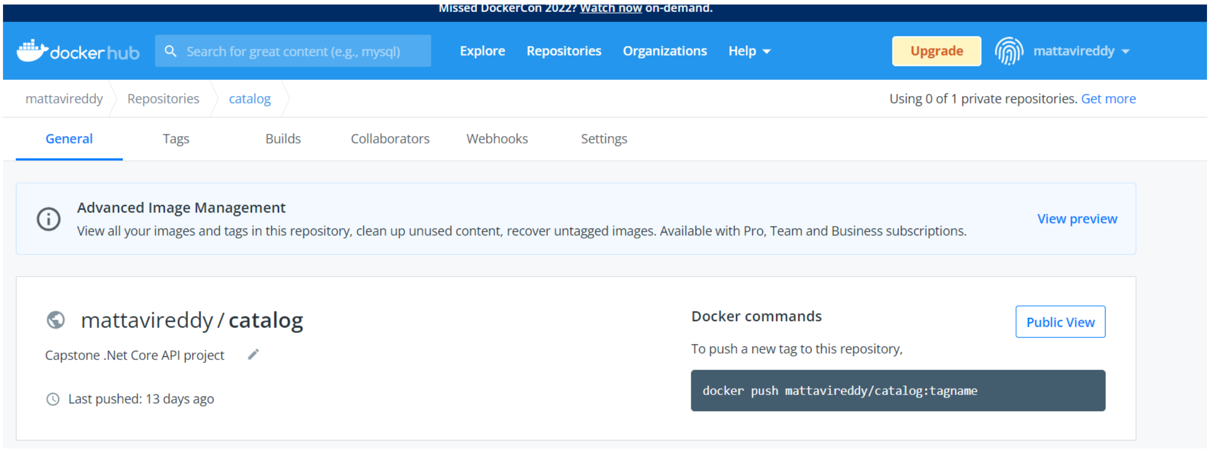
Capstone UI React project

Last pushed: 2 days ago

Docker commands

To push a new tag to this repository,

```
docker push mattavireddy/catalogitems:tagname
```



Using 0 of 1 private repositories.

Advanced Image Management

View all your images and tags in this repository, clean up unused content, recover untagged images. Available with Pro, Team and Business subscriptions.

mattavireddy / **catalog**

Capstone .Net Core API project

Last pushed: 13 days ago

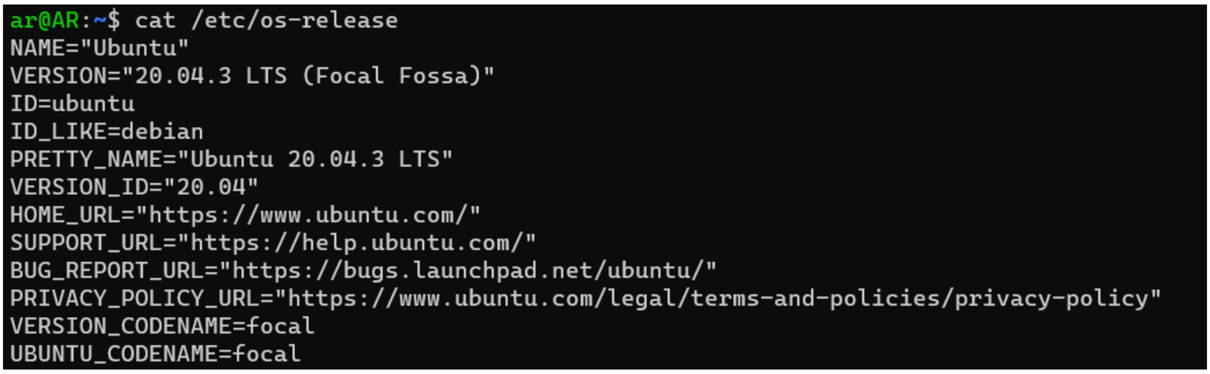
Docker commands

To push a new tag to this repository,

```
docker push mattavireddy/catalog:tagname
```

1. Installation and configuration of required packages/tools on Machine

Using an Ubuntu 20.04 as base OS

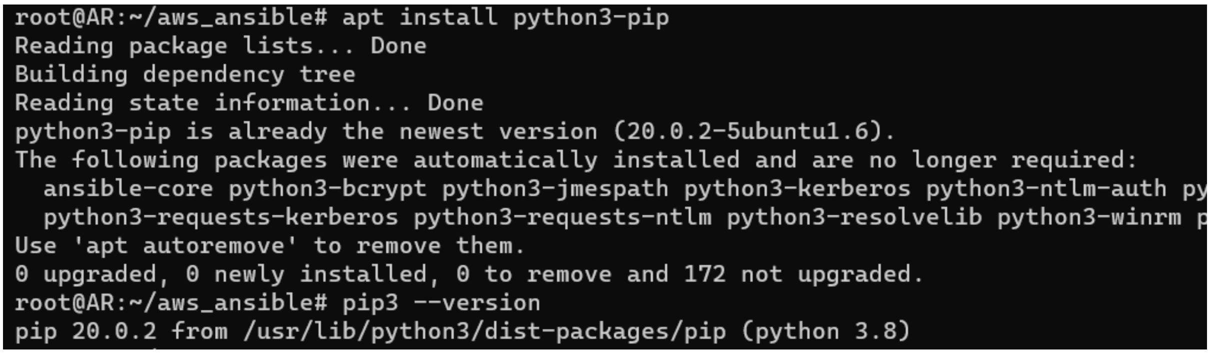


```
ar@AR:~$ cat /etc/os-release
NAME="Ubuntu"
VERSION="20.04.3 LTS (Focal Fossa)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 20.04.3 LTS"
VERSION_ID="20.04"
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
VERSION_CODENAME=focal
UBUNTU_CODENAME=focal
```

Logging In as Sudo User command: **sudo su -**

a) PIP installation

Command: **apt install python3-pip**



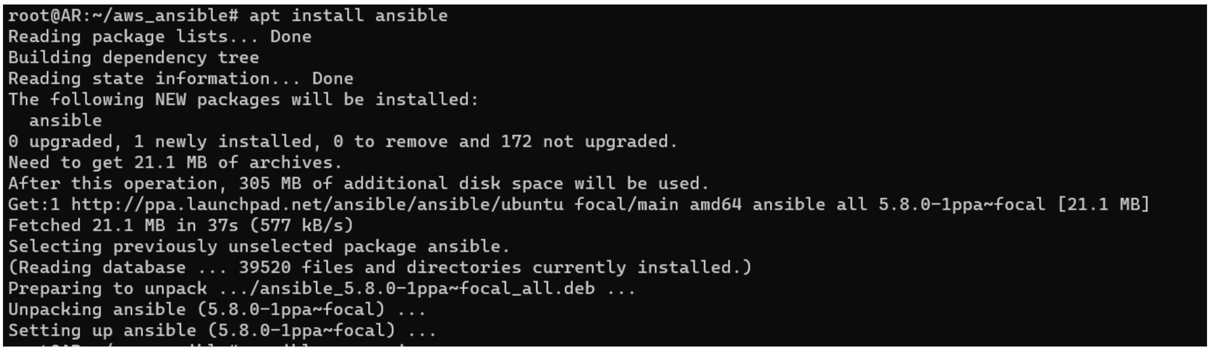
```
root@AR:~/aws_ansible# apt install python3-pip
Reading package lists... Done
Building dependency tree
Reading state information... Done
python3-pip is already the newest version (20.0.2-5ubuntu1.6).
The following packages were automatically installed and are no longer required:
  ansible-core python3-bcrypt python3-jmespath python3-kerberos python3-ntlm-auth py
  python3-requests-kerberos python3-requests-ntlm python3-resolvelib python3-winrm p
Use 'apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 172 not upgraded.
root@AR:~/aws_ansible# pip3 --version
pip 20.0.2 from /usr/lib/python3/dist-packages/pip (python 3.8)
```

b) Ansible Installation

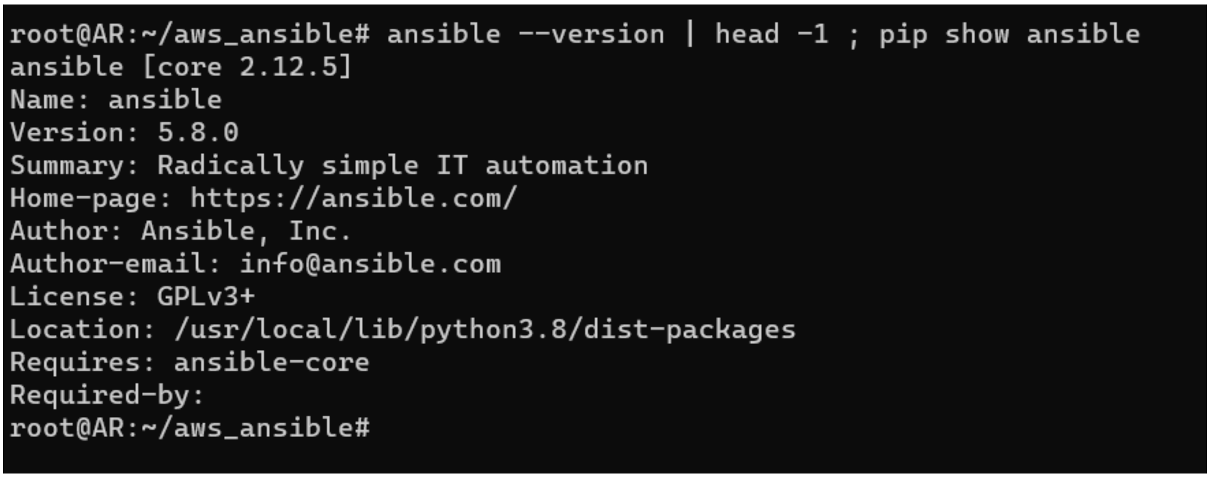
An automation tool for software provisioning, configuration management, and application-deployment tool enabling infrastructure as code.

Command: **apt install ansible**

Verify Version: **ansible --version | head -1; pip show ansible**



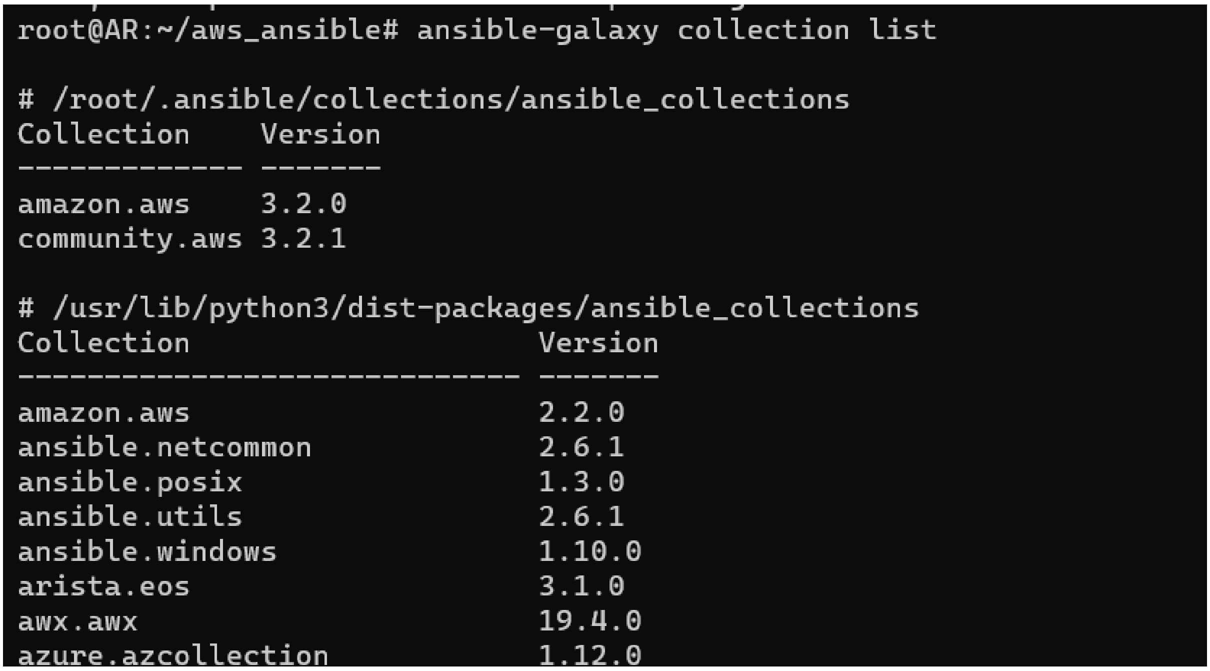
```
root@AR:~/aws_ansible# apt install ansible
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  ansible
0 upgraded, 1 newly installed, 0 to remove and 172 not upgraded.
Need to get 21.1 MB of archives.
After this operation, 305 MB of additional disk space will be used.
Get:1 http://ppa.launchpad.net/ansible/ansible/ubuntu focal/main amd64 ansible all 5.8.0-1ppa~focal [21.1 MB]
Fetched 21.1 MB in 37s (577 kB/s)
Selecting previously unselected package ansible.
(Reading database ... 39520 files and directories currently installed.)
Preparing to unpack .../ansible_5.8.0-1ppa~focal_all.deb ...
Unpacking ansible (5.8.0-1ppa~focal) ...
Setting up ansible (5.8.0-1ppa~focal) ...
```



```
root@AR:~/aws_ansible# ansible --version | head -1 ; pip show ansible
ansible [core 2.12.5]
Name: ansible
Version: 5.8.0
Summary: Radically simple IT automation
Home-page: https://ansible.com/
Author: Ansible, Inc.
Author-email: info@ansible.com
License: GPLv3+
Location: /usr/local/lib/python3.8/dist-packages
Requires: ansible-core
Required-by:
root@AR:~/aws_ansible#
```

c) Aws-collection ansible

Ansible amazon.aws collection helps to configure AWS cloud resources.
Command: **ansible-galaxy collection install amazon.aws**
Verify: **ansible-galaxy collection list**
Source:
**https://docs.ansible.com/ansible/latest/collections/amazon/aws/index.html#plugins-in-amazon-aws**



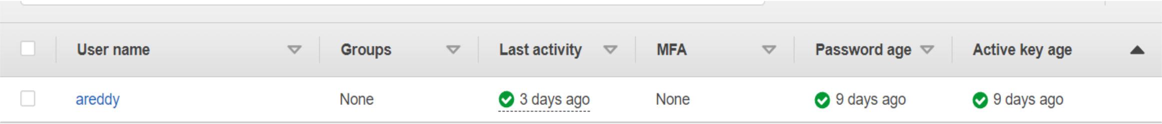
```
root@AR:~/aws_ansible# ansible-galaxy collection list

# /root/.ansible/collections/ansible_collections
Collection    Version
------------- -------
amazon.aws    3.2.0
community.aws 3.2.1

# /usr/lib/python3/dist-packages/ansible_collections
Collection                    Version
----------------------------- -------
amazon.aws                    2.2.0
ansible.netcommon             2.6.1
ansible.posix                 1.3.0
ansible.utils                 2.6.1
ansible.windows               1.10.0
arista.eos                    3.1.0
awx.awx                       19.4.0
azure.azcollection            1.12.0
```

2. AWS account configuration

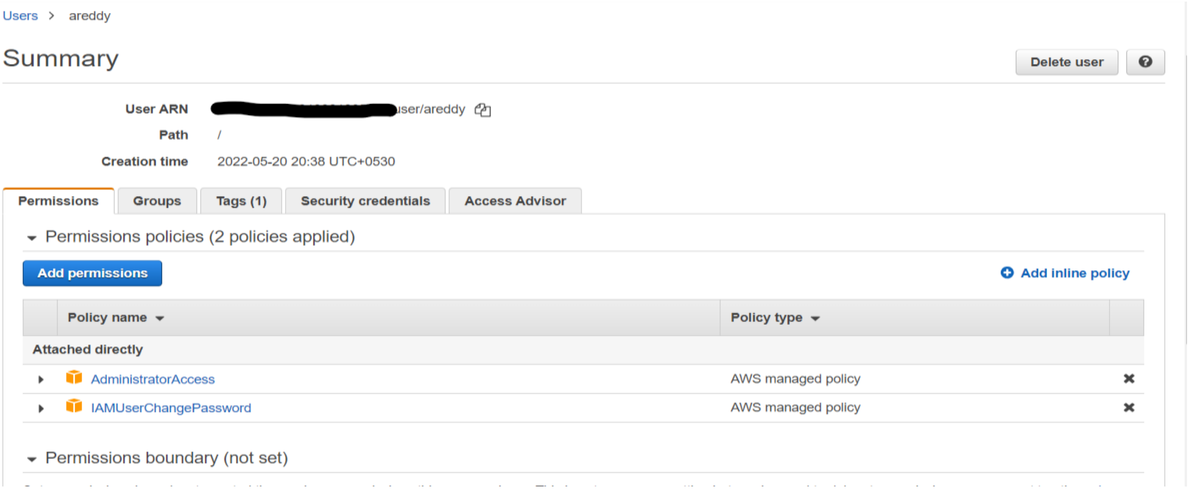
a) IAM User setup

Create an IAM username: **areddy** with AdministratorAccess Policy attached

Using Access_key, Secret_key to create resources on AWS Cloud via Ansible



| □ | User name | Groups | Last activity | MFA | Password age | Active key age |
|---|---|---|---|---|---|---|
| □ | areddy | None | ✓ 3 days ago | None | ✓ 9 days ago | ✓ 9 days ago |



Users > areddy

Summary

User ARN [redacted]ser/areddy

Path /

Creation time 2022-05-20 20:38 UTC+0530

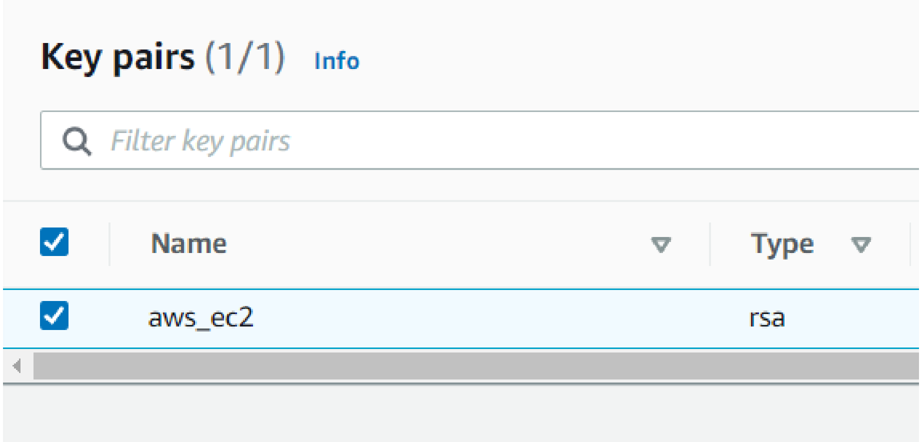
▾ Permissions policies (2 policies applied)

| Policy name | Policy type | |
|---|---|---|
| Attached directly | | |
| ▸ AdministratorAccess | AWS managed policy | ✖ |
| ▸ IAMUserChangePassword | AWS managed policy | ✖ |

▾ Permissions boundary (not set)

b) Key Pair setup

Create a new key pair ‘**aws_ec2**’ for SSH into EC2 instances via Ansible and download the aws_ec2.pem file



Key pairs (1/1) Info

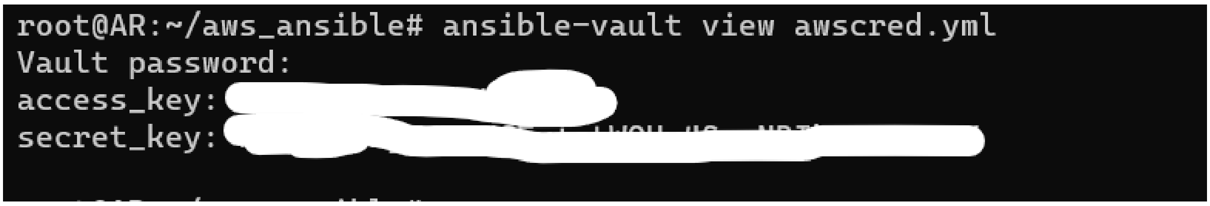
| ☑ | Name | Type |
|---|---|---|
| ☑ | aws_ec2 | rsa |

3. Ansible configuration and scripts files

a) Create directory working space  
Command: **mkdir aws_ansible**  
Commnad: **cd aws_ansible**

b) Create ansible vault file

Storing **IAM** user **‘areddy’** access_key, secret_key into ansible vault file  
Command: **ansible-vault create awscred.yml**  
Command: **ansibel-vault view awscred.yml**



```
root@AR:~/aws_ansible# ansible-vault view awscred.yml
Vault password:
access_key: [redacted]
secret_key: [redacted]
```

c) Create Ansible Roles  
Under new directory roles, creating roles, each role play execute an specific task of our automation process ( each role is described in later pages )

Commands:

**mkdir roles**

**cd roles**

**ansible-galaxy init ec2**

**ansible-galaxy init loadbalancer**

**ansible-galaxy init elastic_ip_address**

**ansible-galaxy init k8s_master**

**ansible-galaxy init k8s_slave**

**ansible-galaxy init pods**

d) Create Ansible config file

Create the ansible.cfg file in the root working space directory ‘aws_ansible’.
Command: **vi ansible.cfg**

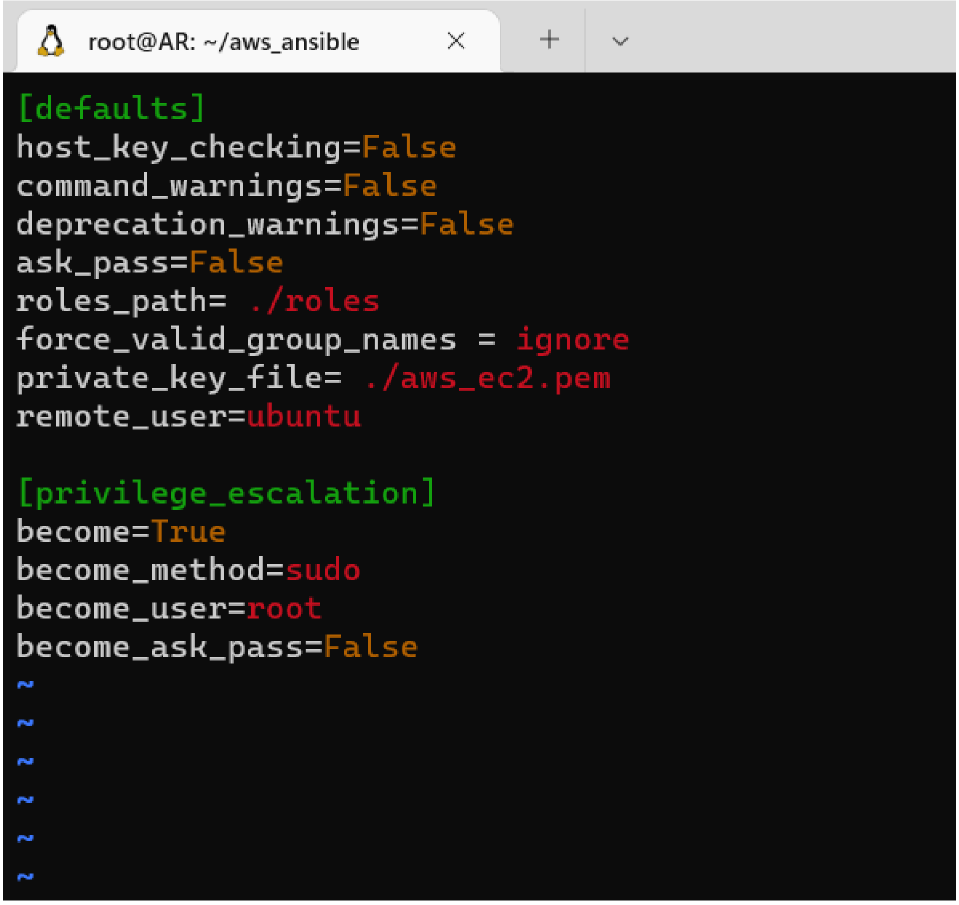
**host_key_checking**: is false, because ec2 instances are connected via SSH by anisble.

**roles_path**: with respect to ‘aws_ansible’ directory pointing out roles path to created roles directory.

**remote_user:** EC2 UBUNTU Instance remote user is "ubuntu" by default.
Source: [https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/connection-prereqs.html](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/connection-prereqs.html)

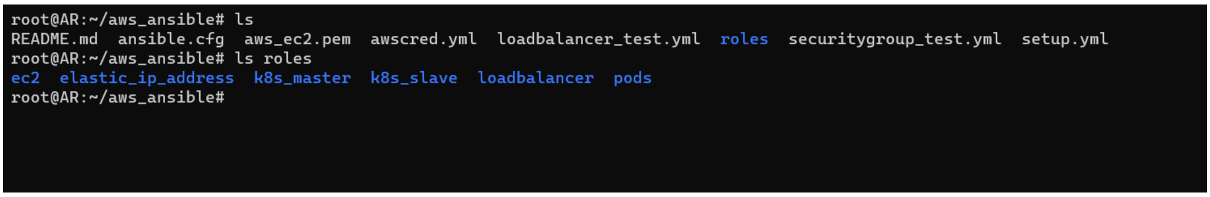
**private_key_file:** path of key value pair downloaded from AWS.

**privilege_escalation:** running commands via ansible as root user**.**



```
[defaults]
host_key_checking=False
command_warnings=False
deprecation_warnings=False
ask_pass=False
roles_path= ./roles
force_valid_group_names = ignore
private_key_file= ./aws_ec2.pem
remote_user=ubuntu

[privilege_escalation]
become=True
become_method=sudo
become_user=root
become_ask_pass=False
~
~
~
~
~
~
```

e) Look at Working directory



```
root@AR:~/aws_ansible# ls
README.md  ansible.cfg  aws_ec2.pem  awscred.yml  loadbalancer_test.yml  roles  securitygroup_test.yml  setup.yml
root@AR:~/aws_ansible# ls roles
ec2  elastic_ip_address  k8s_master  k8s_slave  loadbalancer  pods
root@AR:~/aws_ansible#
```

4. Ansible Roles Overview

a) EC2 Role

Responsible for creating a security group, creating instances (divided into groups, first one into master, rest are into slave) and attaching these instances to created security group

EC2 role vars main.file

**instance_tag:** An array of Items indicating how many EC2 instances to be created

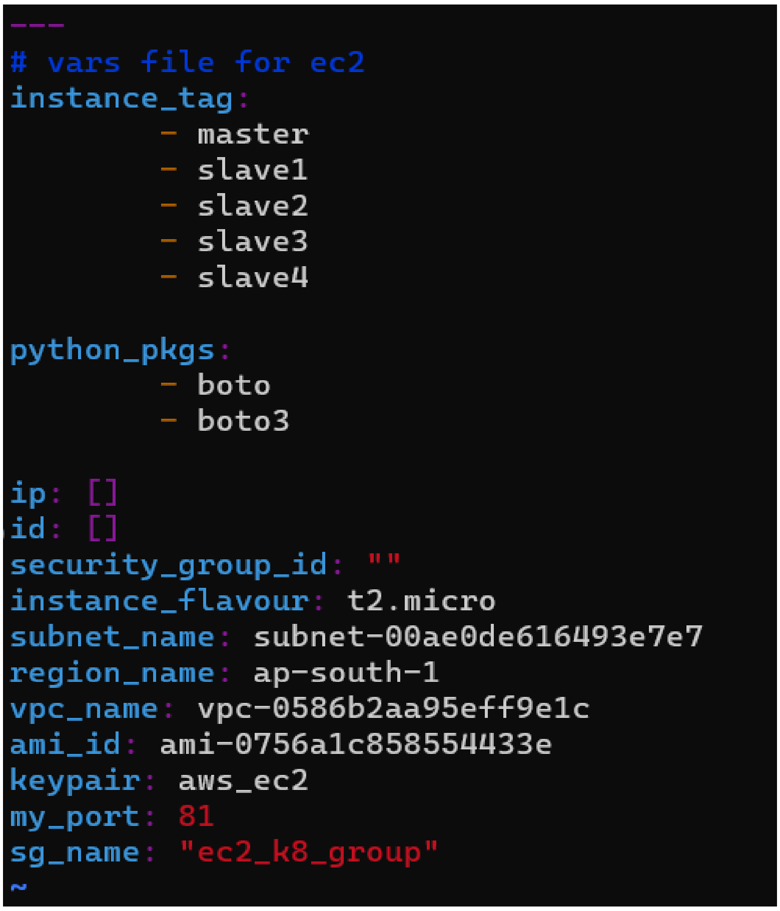
**python_pkgs:** Dependencies to install, for ansible to work with AWS

**sg_name:** Security Group Name

**keypair:** pointing to key pair at AWS

**region_name:** ap-south-1 (Mumbai)

id, ip array holds the data of EC2 instances id and ip address using ansible **'set_fact'**

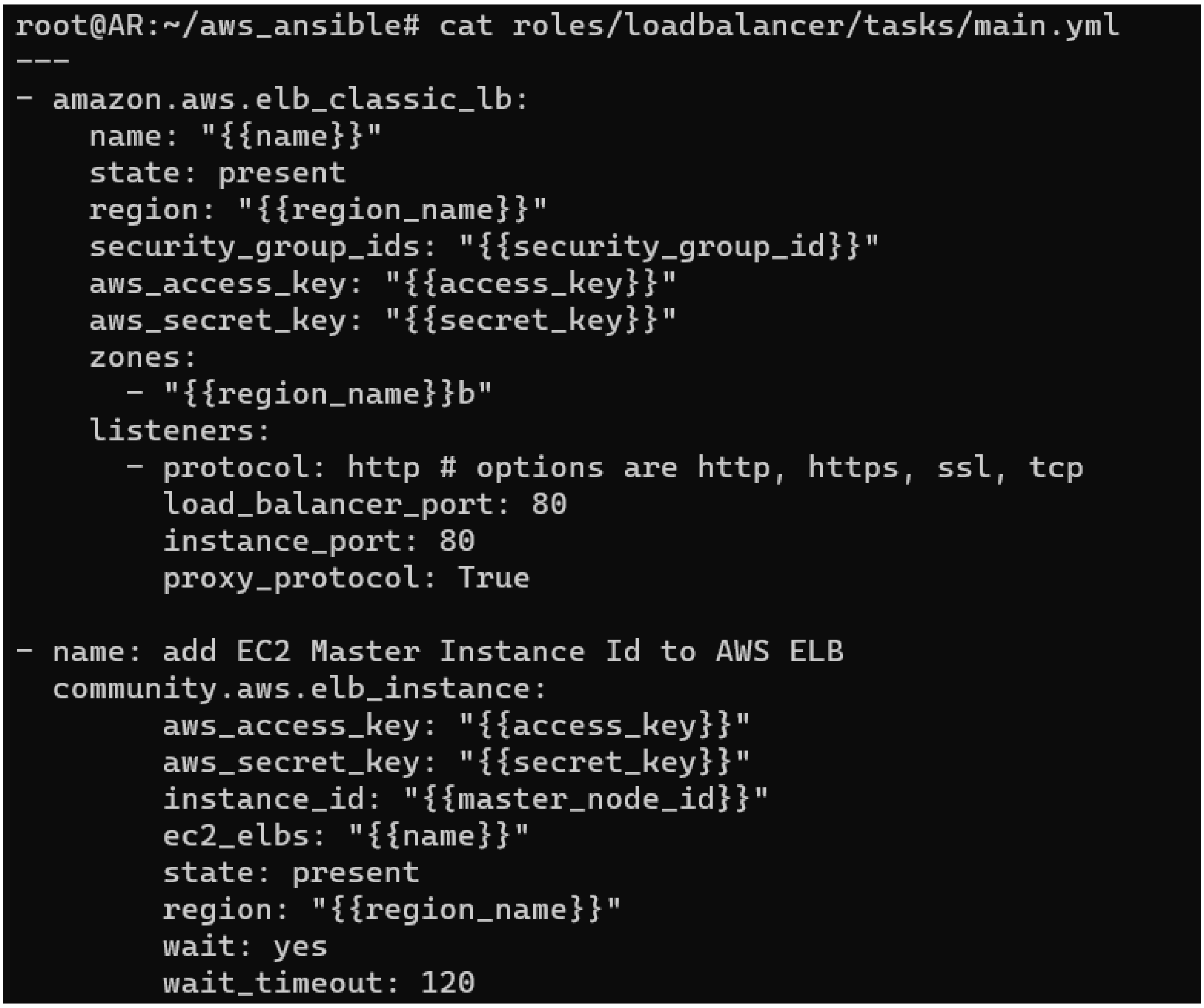


```
---
# vars file for ec2
instance_tag:
        - master
        - slave1
        - slave2
        - slave3
        - slave4

python_pkgs:
        - boto
        - boto3

ip: []
id: []
security_group_id: ""
instance_flavour: t2.micro
subnet_name: subnet-00ae0de616493e7e7
region_name: ap-south-1
vpc_name: vpc-0586b2aa95eff9e1c
ami_id: ami-0756a1c858554433e
keypair: aws_ec2
my_port: 81
sg_name: "ec2_k8_group"
~
```

b) LoadBalancer Role

Responsible for creating a loadbalancer and attaches only master node to it (ingress-nginx-controller service is on master node, it takes care of loading endpoint via path of ingress services)

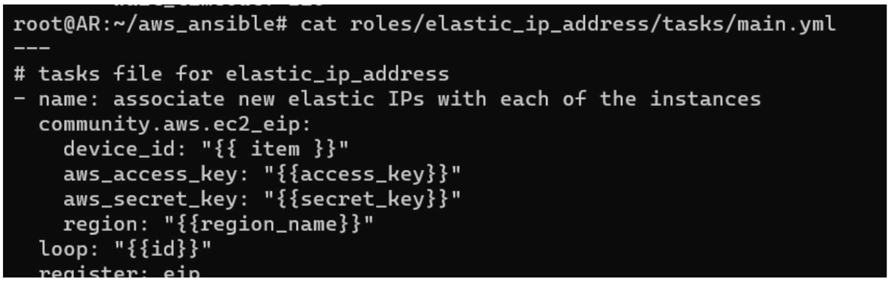
**Note:** can add other EC2 instances but no use here.



```
root@AR:~/aws_ansible# cat roles/loadbalancer/tasks/main.yml
---
- amazon.aws.elb_classic_lb:
    name: "{{name}}"
    state: present
    region: "{{region_name}}"
    security_group_ids: "{{security_group_id}}"
    aws_access_key: "{{access_key}}"
    aws_secret_key: "{{secret_key}}"
    zones:
      - "{{region_name}}b"
    listeners:
      - protocol: http # options are http, https, ssl, tcp
        load_balancer_port: 80
        instance_port: 80
        proxy_protocol: True

- name: add EC2 Master Instance Id to AWS ELB
  community.aws.elb_instance:
        aws_access_key: "{{access_key}}"
        aws_secret_key: "{{secret_key}}"
        instance_id: "{{master_node_id}}"
        ec2_elbs: "{{name}}"
        state: present
        region: "{{region_name}}"
        wait: yes
        wait_timeout: 120
```

c) Elastic_IP_Address Role

Responsible for creating elastic ip address for each instance created.

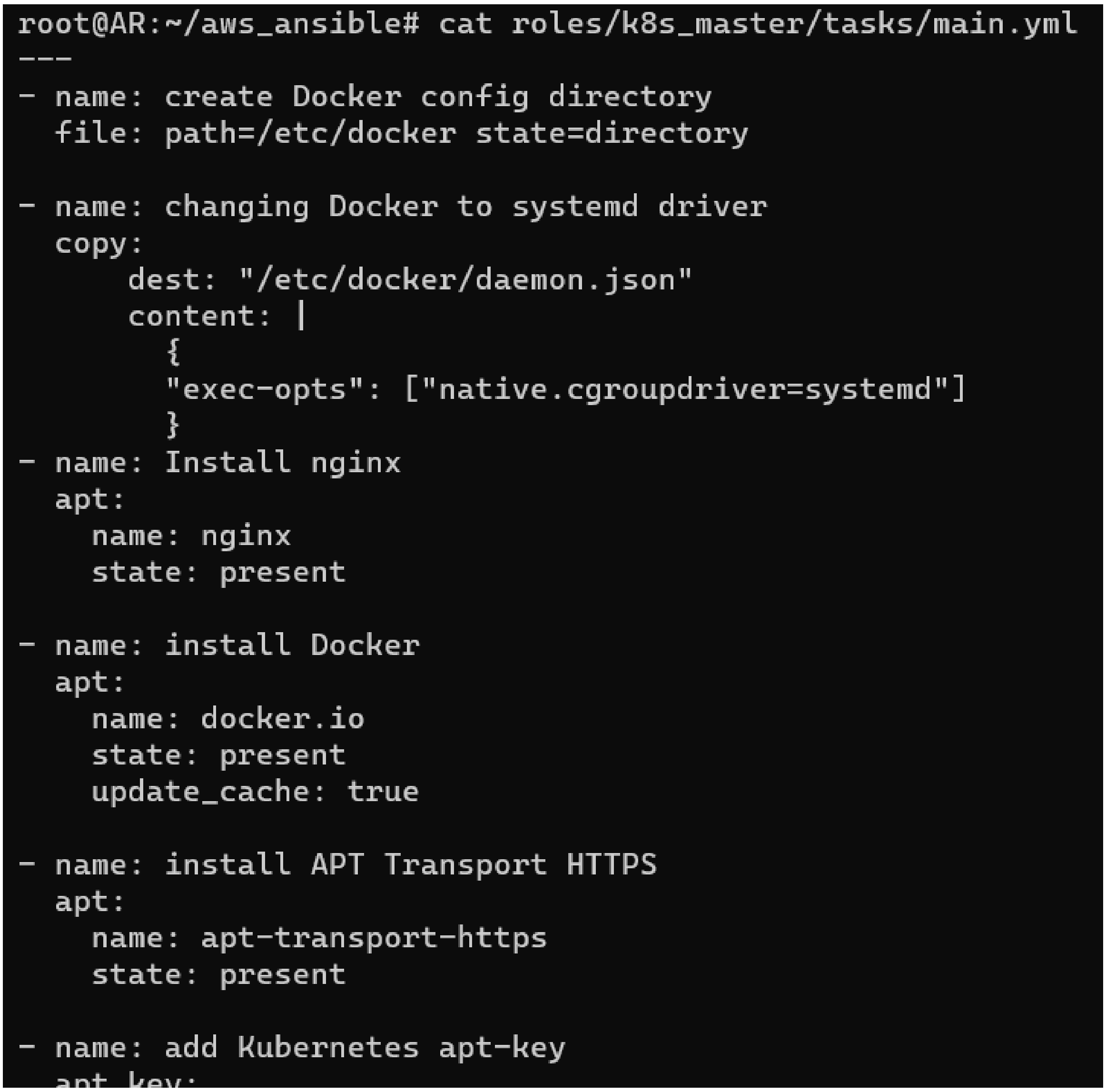


```
root@AR:~/aws_ansible# cat roles/elastic_ip_address/tasks/main.yml
---
# tasks file for elastic_ip_address
- name: associate new elastic IPs with each of the instances
  community.aws.ec2_eip:
    device_id: "{{ item }}"
    aws_access_key: "{{access_key}}"
    aws_secret_key: "{{secret_key}}"
    region: "{{region_name}}"
  loop: "{{id}}"
  register: eip
```

d) K8s_Master Role

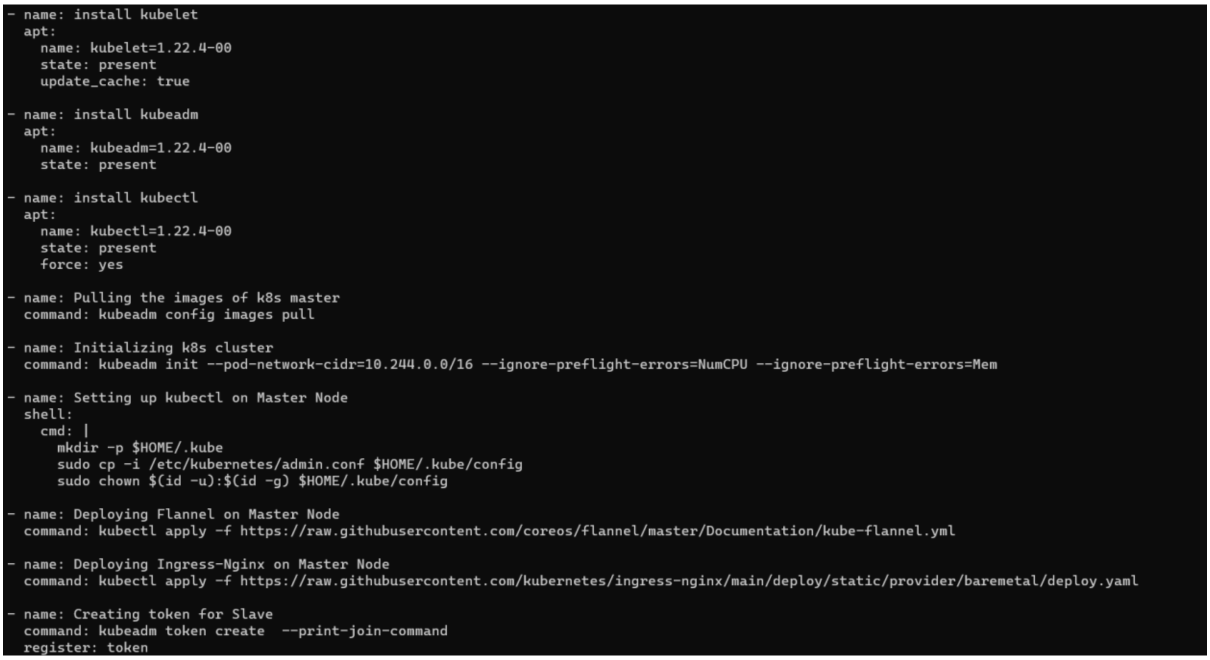
Responsible for installation of Docker and K8 cluster required configurations on Master Node,  and Initializes K8 Cluster setup (kubeadm init) and installs Flannel Network, Ingress-Nginx controller on master node.

Installation of Docker:



```
root@AR:~/aws_ansible# cat roles/k8s_master/tasks/main.yml
---
- name: create Docker config directory
  file: path=/etc/docker state=directory

- name: changing Docker to systemd driver
  copy:
      dest: "/etc/docker/daemon.json"
      content: |
        {
        "exec-opts": ["native.cgroupdriver=systemd"]
        }
- name: Install nginx
  apt:
    name: nginx
    state: present

- name: install Docker
  apt:
    name: docker.io
    state: present
    update_cache: true

- name: install APT Transport HTTPS
  apt:
    name: apt-transport-https
    state: present

- name: add Kubernetes apt-key
  apt_key:
```

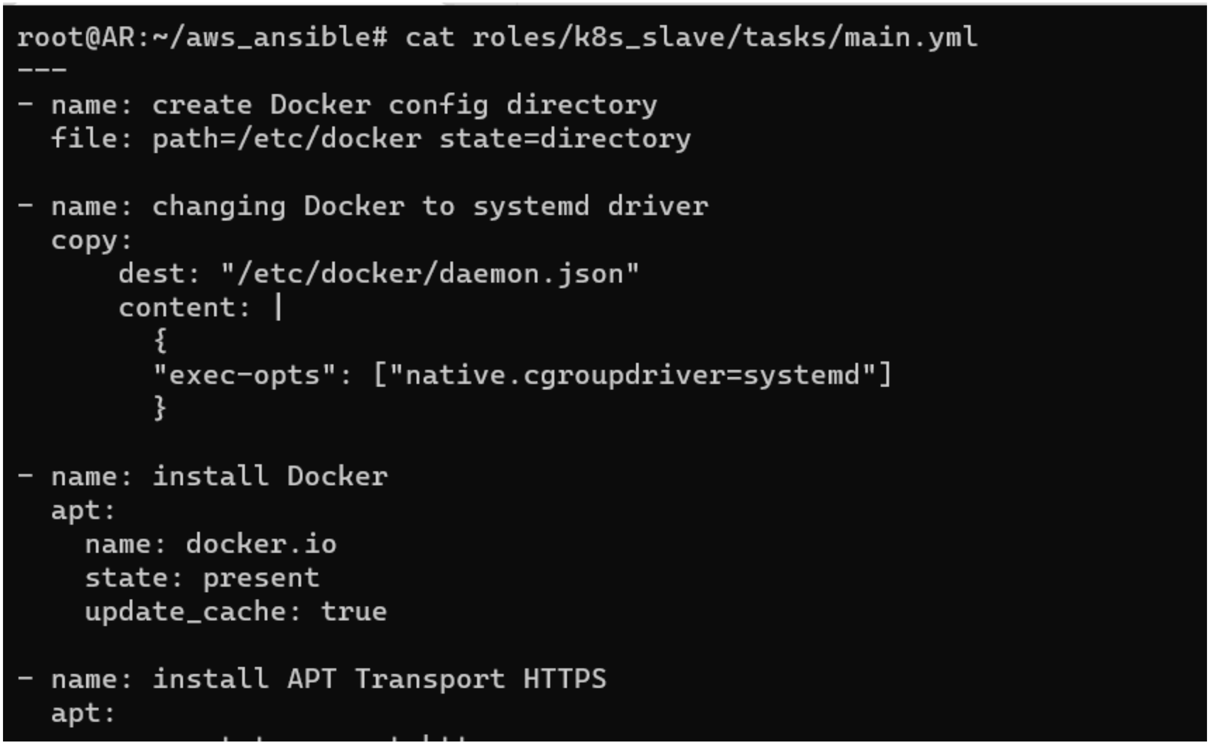
Installation of K8 version V1.22.4, flannel network, Ingress-Nginx Controller,



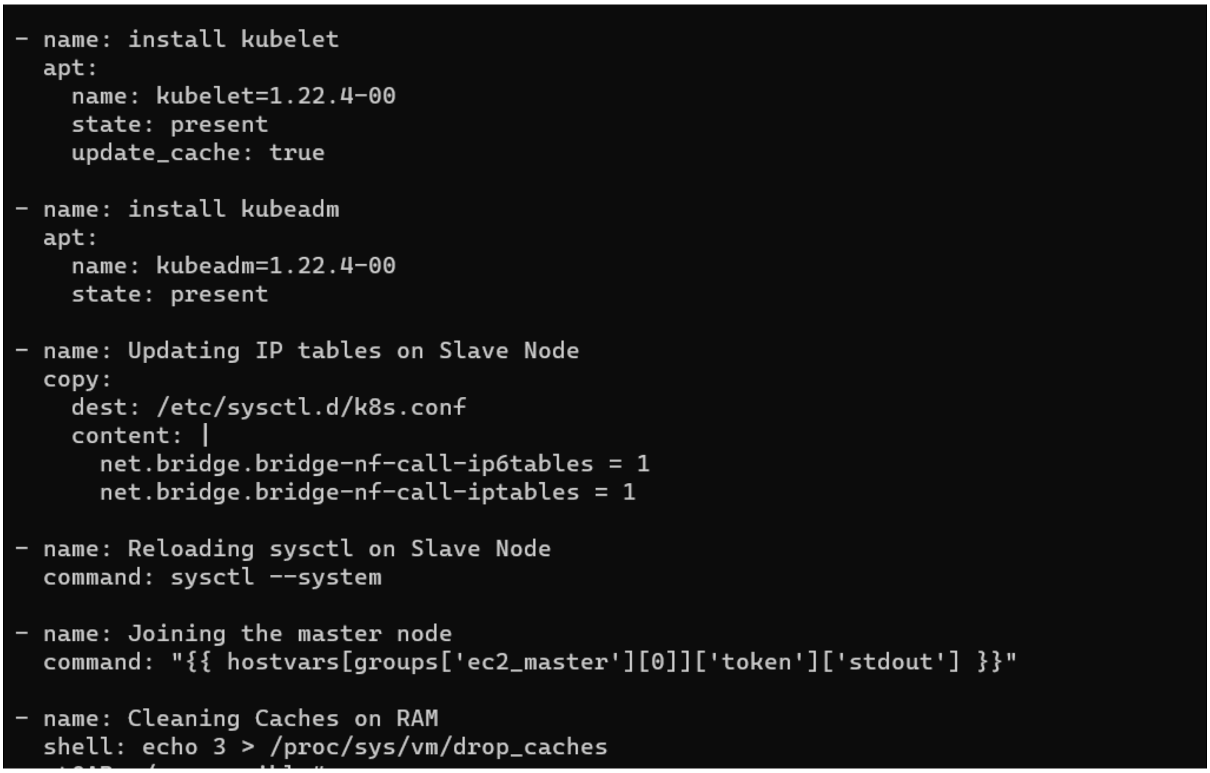
```
- name: install kubelet
  apt:
    name: kubelet=1.22.4-00
    state: present
    update_cache: true

- name: install kubeadm
  apt:
    name: kubeadm=1.22.4-00
    state: present

- name: install kubectl
  apt:
    name: kubectl=1.22.4-00
    state: present
    force: yes

- name: Pulling the images of k8s master
  command: kubeadm config images pull

- name: Initializing k8s cluster
  command: kubeadm init --pod-network-cidr=10.244.0.0/16 --ignore-preflight-errors=NumCPU --ignore-preflight-errors=Mem

- name: Setting up kubectl on Master Node
  shell:
    cmd: |
      mkdir -p $HOME/.kube
      sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
      sudo chown $(id -u):$(id -g) $HOME/.kube/config

- name: Deploying Flannel on Master Node
  command: kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml

- name: Deploying Ingress-Nginx on Master Node
  command: kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/main/deploy/static/provider/baremetal/deploy.yaml

- name: Creating token for Slave
  command: kubeadm token create  --print-join-command
  register: token
```

e) K8s_Slave Role

Responsible for installation of Docker and K8 cluster required configurations on slave nodes, and joins with above master node



```
root@AR:~/aws_ansible# cat roles/k8s_slave/tasks/main.yml
---
- name: create Docker config directory
  file: path=/etc/docker state=directory

- name: changing Docker to systemd driver
  copy:
      dest: "/etc/docker/daemon.json"
      content: |
        {
        "exec-opts": ["native.cgroupdriver=systemd"]
        }

- name: install Docker
  apt:
    name: docker.io
    state: present
    update_cache: true

- name: install APT Transport HTTPS
  apt:
```



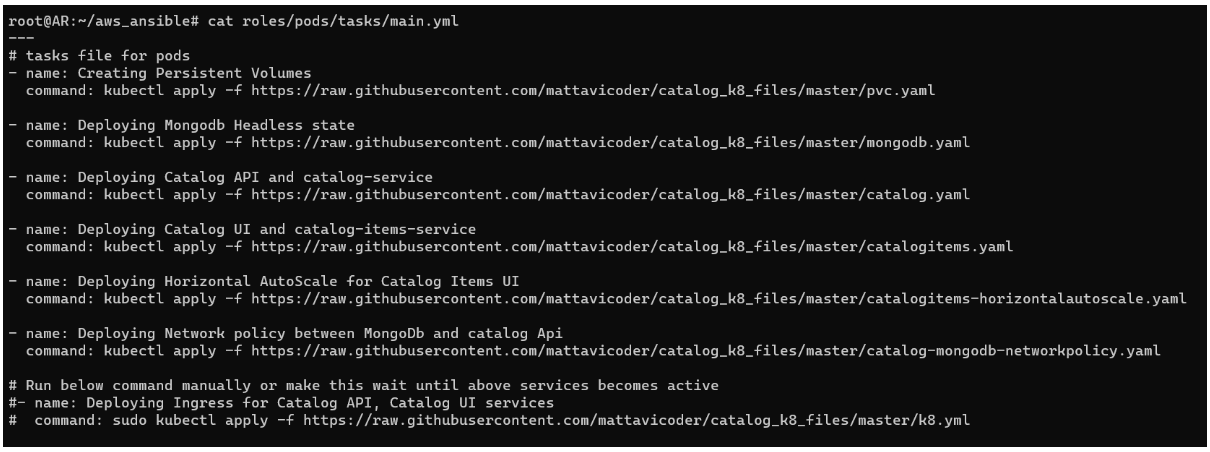
```
- name: install kubelet
  apt:
    name: kubelet=1.22.4-00
    state: present
    update_cache: true

- name: install kubeadm
  apt:
    name: kubeadm=1.22.4-00
    state: present

- name: Updating IP tables on Slave Node
  copy:
    dest: /etc/sysctl.d/k8s.conf
    content: |
      net.bridge.bridge-nf-call-ip6tables = 1
      net.bridge.bridge-nf-call-iptables = 1

- name: Reloading sysctl on Slave Node
  command: sysctl --system

- name: Joining the master node
  command: "{{ hostvars[groups['ec2_master'][0]]['token']['stdout'] }}"

- name: Cleaning Caches on RAM
  shell: echo 3 > /proc/sys/vm/drop_caches
```

f) Pods Roles

Responsible for Deploying Statefulsets (MongoDb), Deployments (API, UI ),

ClusterIP Services( API, UI, MongoDb) , NetworkPolicies (between DB and API),

Ingress services (API, UI), HorizontalPodAutoscaler (UI)



```
root@AR:~/aws_ansible# cat roles/pods/tasks/main.yml
---
# tasks file for pods
- name: Creating Persistent Volumes
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/pvc.yaml

- name: Deploying Mongodb Headless state
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/mongodb.yaml

- name: Deploying Catalog API and catalog-service
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/catalog.yaml

- name: Deploying Catalog UI and catalog-items-service
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/catalogitems.yaml

- name: Deploying Horizontal AutoScale for Catalog Items UI
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/catalogitems-horizontalautoscale.yaml

- name: Deploying Network policy between MongoDb and catalog Api
  command: kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/catalog-mongodb-networkpolicy.yaml

# Run below command manually or make this wait until above services becomes active
#- name: Deploying Ingress for Catalog API, Catalog UI services
#  command: sudo kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/k8.yml
```

5. Source of Ansible Scripts and K8 objects files

a) Ansible Scripts GitHub Link

Source: mattavicoder/aws_ec2_k8_ansible_scripts: Creating AWS EC2 instances and configuration of K8 Cluster setup and running catalog Docker repos (github.com)

b) K8 Objects GitHub Link

Source: GitHub - mattavicoder/catalog_k8_files: Capstone Project Yaml files to deploy to Kubernetes

6. Ansible Scripts Execution

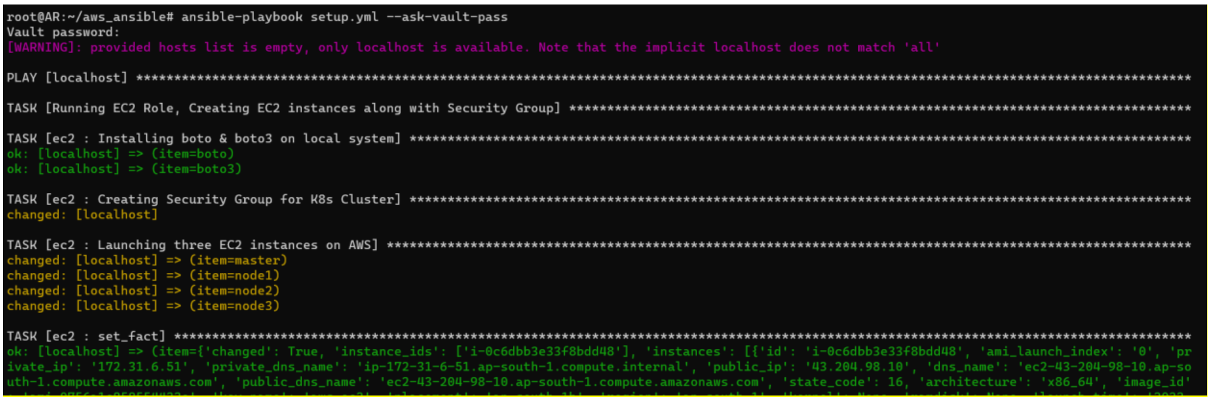
a) Create a Playbook to execute all roles

setup.yml is our playbook ( aws_ec2_k8_ansible_scripts/setup.yml at main · mattavicoder/aws_ec2_k8_ansible_scripts (github.com))

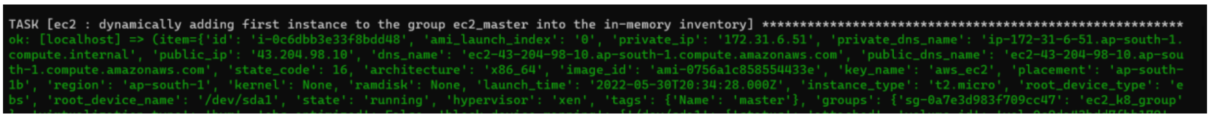
which executes the roles in the following order   ec2, k8s_master, loadbalancer, k8s_slave, pods, elastic_ip_address

command: **ansible-playbook setup.yml --ask-vault-pass**

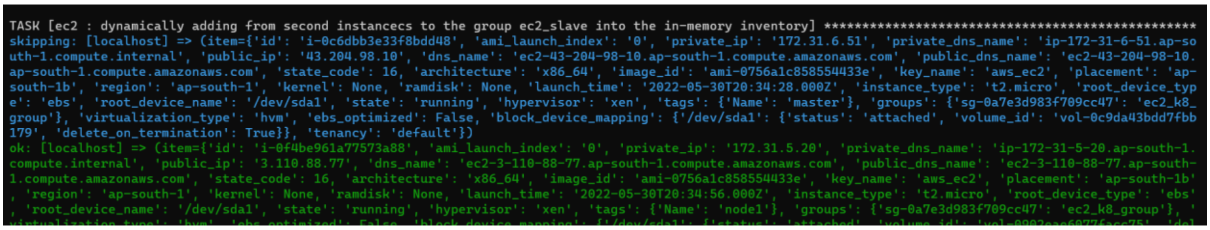
b)  EC2 role results



```
root@AR:~/aws_ansible# ansible-playbook setup.yml --ask-vault-pass
Vault password:
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] ******************************************************************************************************************************

TASK [Running EC2 Role, Creating EC2 instances along with Security Group] *********************************************************************

TASK [ec2 : Installing boto & boto3 on local system] ******************************************************************************************
ok: [localhost] => (item=boto)
ok: [localhost] => (item=boto3)

TASK [ec2 : Creating Security Group for K8s Cluster] ******************************************************************************************
changed: [localhost]

TASK [ec2 : Launching three EC2 instances on AWS] *********************************************************************************************
changed: [localhost] => (item=master)
changed: [localhost] => (item=node1)
changed: [localhost] => (item=node2)
changed: [localhost] => (item=node3)

TASK [ec2 : set_fact] *************************************************************************************************************************
ok: [localhost] => (item={'changed': True, 'instance_ids': ['i-0c6dbb3e33f8bdd48'], 'instances': [{'id': 'i-0c6dbb3e33f8bdd48', 'ami_launch_index': '0', 'pr
ivate_ip': '172.31.6.51', 'private_dns_name': 'ip-172-31-6-51.ap-south-1.compute.internal', 'public_ip': '43.204.98.10', 'dns_name': 'ec2-43-204-98-10.ap-so
uth-1.compute.amazonaws.com', 'public_dns_name': 'ec2-43-204-98-10.ap-south-1.compute.amazonaws.com', 'state_code': 16, 'architecture': 'x86_64', 'image_id'
```

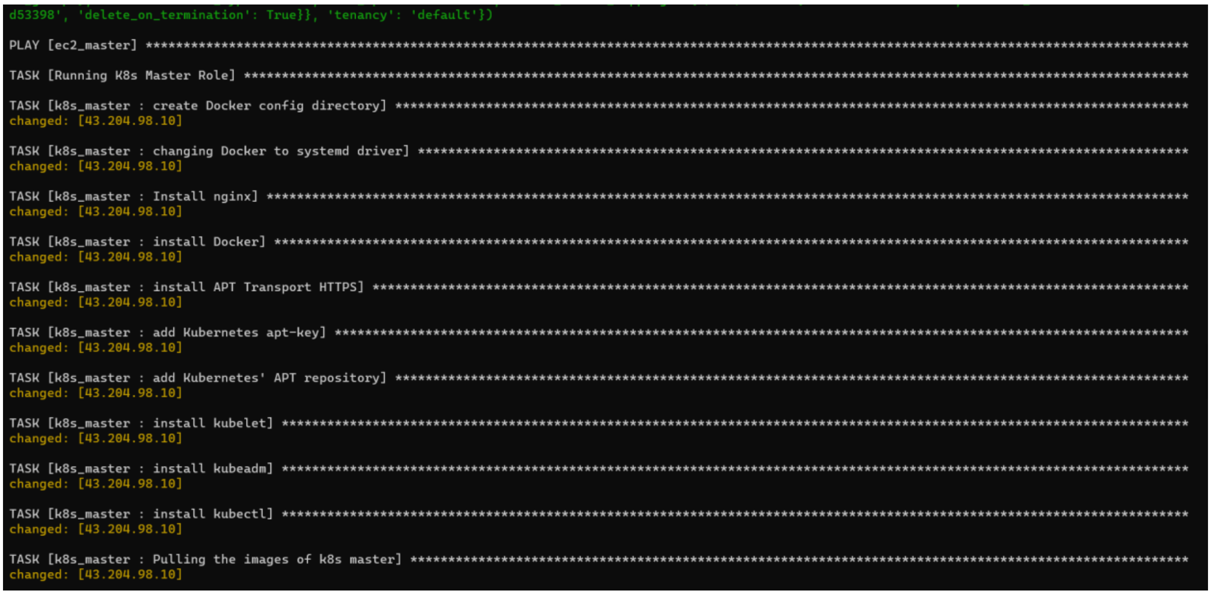


```
TASK [ec2 : dynamically adding first instance to the group ec2_master into the in-memory inventory] *******************************************
ok: [localhost] => (item={'id': 'i-0c6dbb3e33f8bdd48', 'ami_launch_index': '0', 'private_ip': '172.31.6.51', 'private_dns_name': 'ip-172-31-6-51.ap-south-1.
compute.internal', 'public_ip': '43.204.98.10', 'dns_name': 'ec2-43-204-98-10.ap-south-1.compute.amazonaws.com', 'public_dns_name': 'ec2-43-204-98-10.ap-sou
th-1.compute.amazonaws.com', 'state_code': 16, 'architecture': 'x86_64', 'image_id': 'ami-0756a1c858554433e', 'key_name': 'aws_ec2', 'placement': 'ap-south-
1b', 'region': 'ap-south-1', 'kernel': None, 'ramdisk': None, 'launch_time': '2022-05-30T20:34:28.000Z', 'instance_type': 't2.micro', 'root_device_type': 'e
bs', 'root_device_name': '/dev/sda1', 'state': 'running', 'hypervisor': 'xen', 'tags': {'Name': 'master'}, 'groups': {'sg-0a7e3d983f709cc47': 'ec2_k8_group'
```

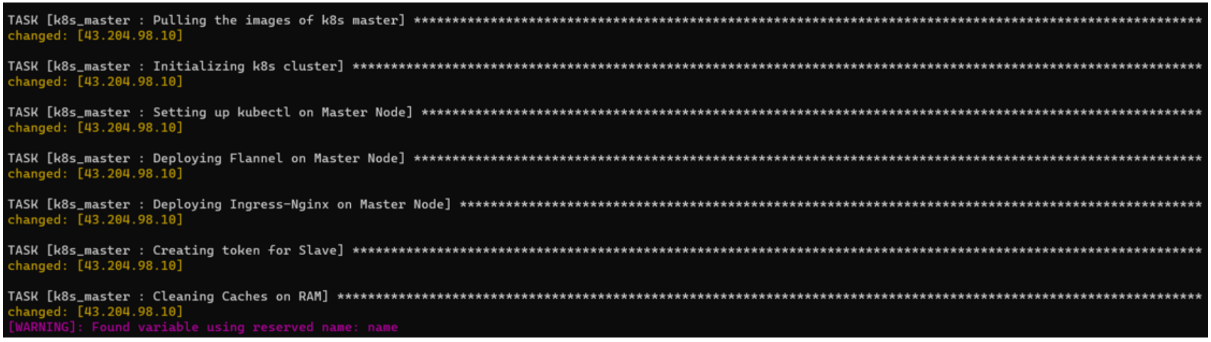


```
TASK [ec2 : dynamically adding from second instancecs to the group ec2_slave into the in-memory inventory] ************************************
skipping: [localhost] => (item={'id': 'i-0c6dbb3e33f8bdd48', 'ami_launch_index': '0', 'private_ip': '172.31.6.51', 'private_dns_name': 'ip-172-31-6-51.ap-so
uth-1.compute.internal', 'public_ip': '43.204.98.10', 'dns_name': 'ec2-43-204-98-10.ap-south-1.compute.amazonaws.com', 'public_dns_name': 'ec2-43-204-98-10.
ap-south-1.compute.amazonaws.com', 'state_code': 16, 'architecture': 'x86_64', 'image_id': 'ami-0756a1c858554433e', 'key_name': 'aws_ec2', 'placement': 'ap-
south-1b', 'region': 'ap-south-1', 'kernel': None, 'ramdisk': None, 'launch_time': '2022-05-30T20:34:28.000Z', 'instance_type': 't2.micro', 'root_device_typ
e': 'ebs', 'root_device_name': '/dev/sda1', 'state': 'running', 'hypervisor': 'xen', 'tags': {'Name': 'master'}, 'groups': {'sg-0a7e3d983f709cc47': 'ec2_k8_
group'}, 'virtualization_type': 'hvm', 'ebs_optimized': False, 'block_device_mapping': {'/dev/sda1': {'status': 'attached', 'volume_id': 'vol-0c9da43bdd7fbb
179', 'delete_on_termination': True}}, 'tenancy': 'default'})
ok: [localhost] => (item={'id': 'i-0f4be961a77573a88', 'ami_launch_index': '0', 'private_ip': '172.31.5.20', 'private_dns_name': 'ip-172-31-5-20.ap-south-1.
compute.internal', 'public_ip': '3.110.88.77', 'dns_name': 'ec2-3-110-88-77.ap-south-1.compute.amazonaws.com', 'public_dns_name': 'ec2-3-110-88-77.ap-south-
1.compute.amazonaws.com', 'state_code': 16, 'architecture': 'x86_64', 'image_id': 'ami-0756a1c858554433e', 'key_name': 'aws_ec2', 'placement': 'ap-south-1b'
, 'region': 'ap-south-1', 'kernel': None, 'ramdisk': None, 'launch_time': '2022-05-30T20:34:56.000Z', 'instance_type': 't2.micro', 'root_device_type': 'ebs'
, 'root_device_name': '/dev/sda1', 'state': 'running', 'hypervisor': 'xen', 'tags': {'Name': 'node1'}, 'groups': {'sg-0a7e3d983f709cc47': 'ec2_k8_group'}, '
```

c) k8s_master Role results

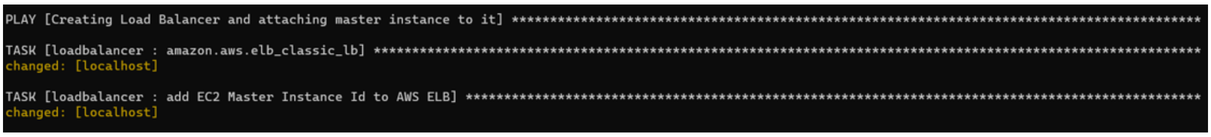


```
d53398', 'delete_on_termination': True}}, 'tenancy': 'default'})

PLAY [ec2_master] ******************************************************************************************************

TASK [Running K8s Master Role] *****************************************************************************************

TASK [k8s_master : create Docker config directory] *********************************************************************
changed: [43.204.98.10]

TASK [k8s_master : changing Docker to systemd driver] ******************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Install nginx] **************************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : install Docker] *************************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : install APT Transport HTTPS] ************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : add Kubernetes apt-key] *****************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : add Kubernetes' APT repository] *********************************************************************
changed: [43.204.98.10]

TASK [k8s_master : install kubelet] ************************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : install kubeadm] ************************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : install kubectl] ************************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Pulling the images of k8s master] *******************************************************************
changed: [43.204.98.10]
```



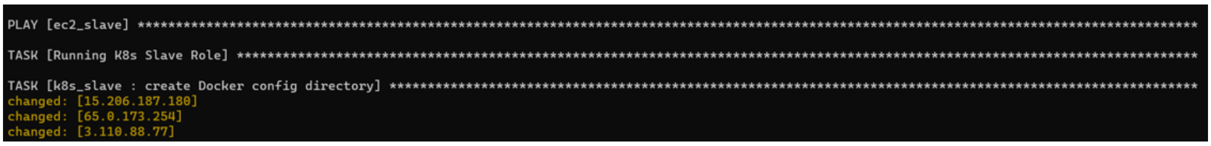
```
TASK [k8s_master : Pulling the images of k8s master] *******************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Initializing k8s cluster] ***************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Setting up kubectl on Master Node] ******************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Deploying Flannel on Master Node] *******************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Deploying Ingress-Nginx on Master Node] *************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Creating token for Slave] ***************************************************************************
changed: [43.204.98.10]

TASK [k8s_master : Cleaning Caches on RAM] *****************************************************************************
changed: [43.204.98.10]
[WARNING]: Found variable using reserved name: name
```

d) LoadBalancer role results

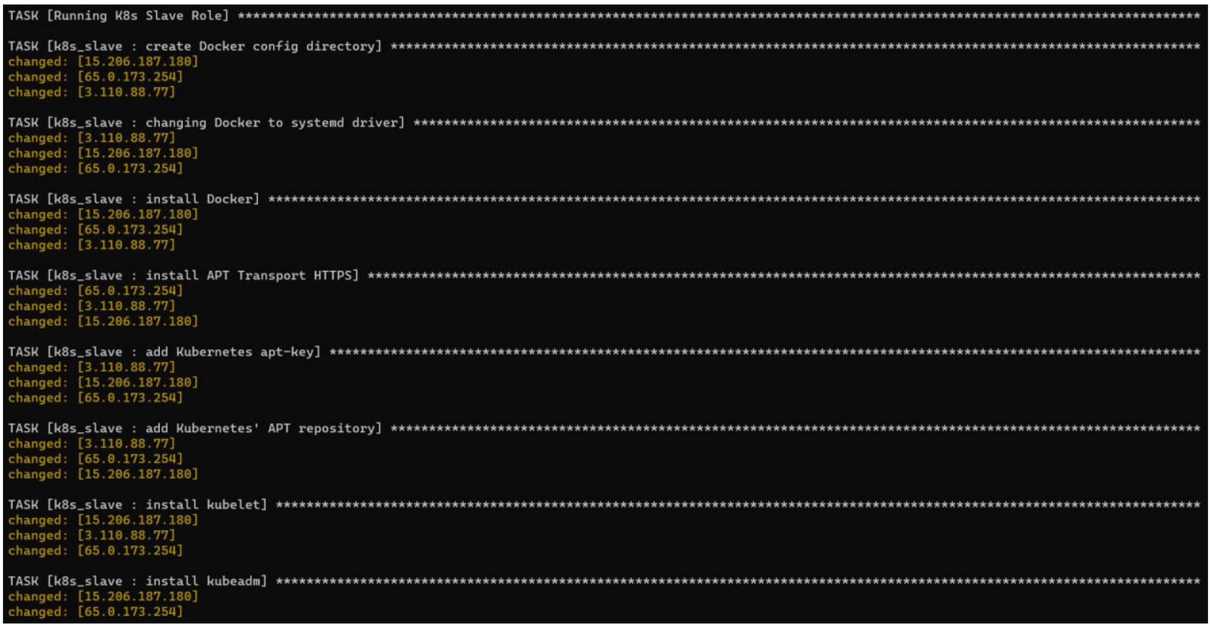


```
PLAY [Creating Load Balancer and attaching master instance to it] ******************************************************

TASK [loadbalancer : amazon.aws.elb_classic_lb] ************************************************************************
changed: [localhost]

TASK [loadbalancer : add EC2 Master Instance Id to AWS ELB] ************************************************************
changed: [localhost]
```

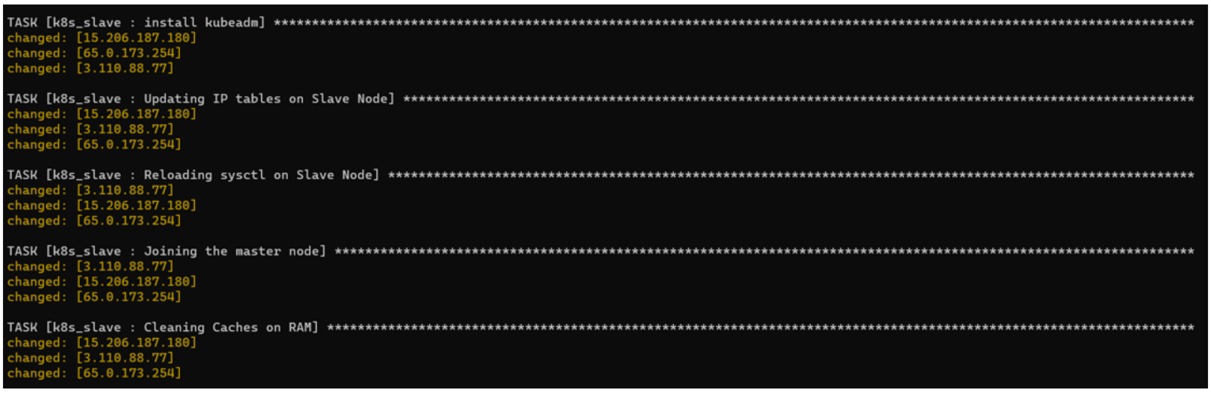
e) K8s_slave role results



```
PLAY [ec2_slave] *******************************************************************************************************

TASK [Running K8s Slave Role] ******************************************************************************************

TASK [k8s_slave : create Docker config directory] **********************************************************************
changed: [15.206.187.180]
changed: [65.0.173.254]
changed: [3.110.88.77]
```



```
TASK [Running K8s Slave Role] ****************************************************************************************************************

TASK [k8s_slave : create Docker config directory] ********************************************************************************************
changed: [15.206.187.180]
changed: [65.0.173.254]
changed: [3.110.88.77]

TASK [k8s_slave : changing Docker to systemd driver] *****************************************************************************************
changed: [3.110.88.77]
changed: [15.206.187.180]
changed: [65.0.173.254]

TASK [k8s_slave : install Docker] ************************************************************************************************************
changed: [15.206.187.180]
changed: [65.0.173.254]
changed: [3.110.88.77]

TASK [k8s_slave : install APT Transport HTTPS] ***********************************************************************************************
changed: [65.0.173.254]
changed: [3.110.88.77]
changed: [15.206.187.180]

TASK [k8s_slave : add Kubernetes apt-key] ****************************************************************************************************
changed: [3.110.88.77]
changed: [15.206.187.180]
changed: [65.0.173.254]

TASK [k8s_slave : add Kubernetes' APT repository] ********************************************************************************************
changed: [3.110.88.77]
changed: [65.0.173.254]
changed: [15.206.187.180]

TASK [k8s_slave : install kubelet] ***********************************************************************************************************
changed: [15.206.187.180]
changed: [3.110.88.77]
changed: [65.0.173.254]

TASK [k8s_slave : install kubeadm] ***********************************************************************************************************
changed: [15.206.187.180]
changed: [65.0.173.254]
```



```
TASK [k8s_slave : install kubeadm] ***********************************************************************************************************
changed: [15.206.187.180]
changed: [65.0.173.254]
changed: [3.110.88.77]

TASK [k8s_slave : Updating IP tables on Slave Node] ******************************************************************************************
changed: [15.206.187.180]
changed: [3.110.88.77]
changed: [65.0.173.254]

TASK [k8s_slave : Reloading sysctl on Slave Node] ********************************************************************************************
changed: [3.110.88.77]
changed: [15.206.187.180]
changed: [65.0.173.254]

TASK [k8s_slave : Joining the master node] ***************************************************************************************************
changed: [3.110.88.77]
changed: [15.206.187.180]
changed: [65.0.173.254]

TASK [k8s_slave : Cleaning Caches on RAM] ****************************************************************************************************
changed: [15.206.187.180]
changed: [3.110.88.77]
changed: [65.0.173.254]
```

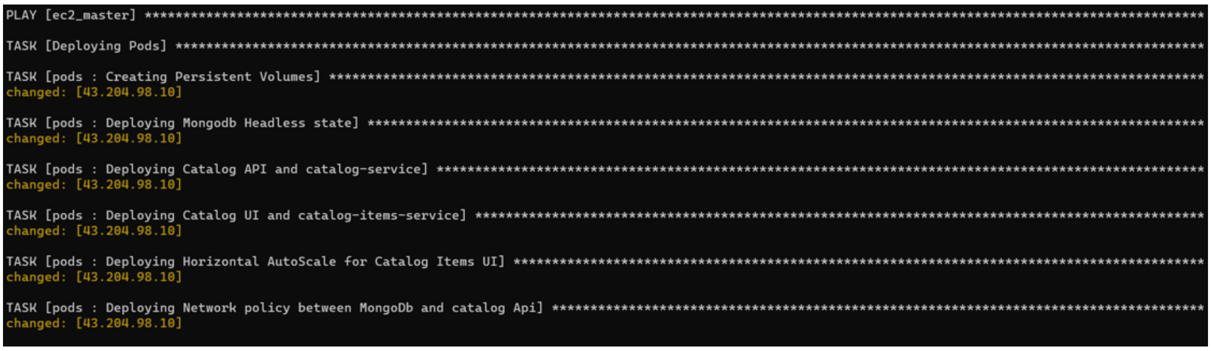
f) Pods role results (deploying k8 objects)

**Note:**

a) Network Policy implemented between MongoDb service and API service GitHub: catalog_k8_files/catalog-mongodb-networkpolicy.yaml at main · mattavicoder/catalog_k8_files (github.com)

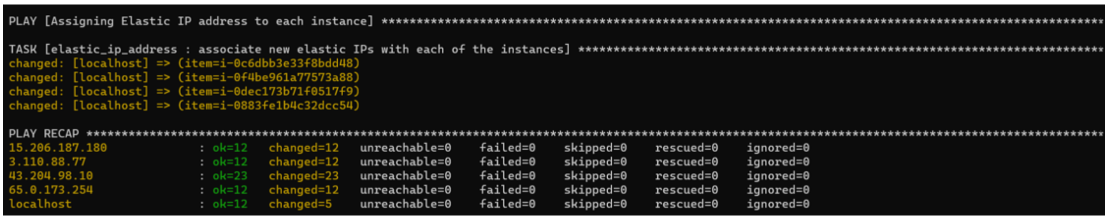
```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: catalog-mongodb-policy
spec:
  podSelector:
    matchLabels:
      role: mongo
  policyTypes:
    - Ingress
  ingress:
    - from:
        - podSelector:
            matchLabels:
              app: catalog
      ports:
        - port: 27017
```

b) Horizontal Auto Scale Object is setup for UI Service.
GitHub: [catalog_k8_files/catalogitems-horizontalautoscale.yaml at main · mattavicoder/catalog_k8_files (github.com)]



```
PLAY [ec2_master] **************************************************************************************************************

TASK [Deploying Pods] **********************************************************************************************************

TASK [pods : Creating Persistent Volumes] **************************************************************************************
changed: [43.204.98.10]

TASK [pods : Deploying Mongodb Headless state] *********************************************************************************
changed: [43.204.98.10]

TASK [pods : Deploying Catalog API and catalog-service] ************************************************************************
changed: [43.204.98.10]

TASK [pods : Deploying Catalog UI and catalog-items-service] *******************************************************************
changed: [43.204.98.10]

TASK [pods : Deploying Horizontal AutoScale for Catalog Items UI] **************************************************************
changed: [43.204.98.10]

TASK [pods : Deploying Network policy between MongoDb and catalog Api] *********************************************************
changed: [43.204.98.10]
```

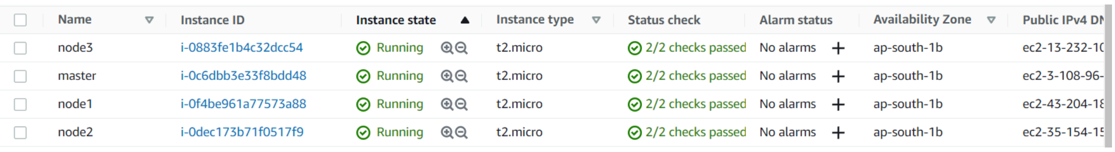
g) Elastic_Ip_Address role results



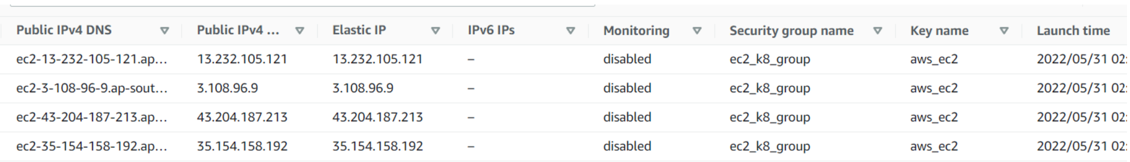
```
PLAY [Assigning Elastic IP address to each instance] ***************************************************************************

TASK [elastic_ip_address : associate new elastic IPs with each of the instances] ***********************************************
changed: [localhost] => (item=i-0c6dbb3e33f8bdd48)
changed: [localhost] => (item=i-0f4be961a77573a88)
changed: [localhost] => (item=i-0dec173b71f0517f9)
changed: [localhost] => (item=i-0883fe1b4c32dcc54)

PLAY RECAP *********************************************************************************************************************
15.206.187.180             : ok=12   changed=12   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
3.110.88.77                : ok=12   changed=12   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
43.204.98.10               : ok=23   changed=23   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
65.0.173.254               : ok=12   changed=12   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
localhost                  : ok=12   changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

## 7. Verify Ansible Results at AWS

### a) EC2 Instances

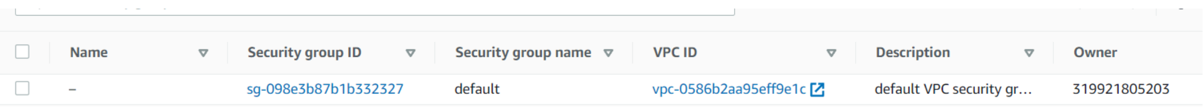


| □ | Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DN |
|---|---|---|---|---|---|---|---|---|
| □ | node3 | i-0883fe1b4c32dcc54 | ✓ Running | t2.micro | ✓ 2/2 checks passed | No alarms + | ap-south-1b | ec2-13-232-1 |
| □ | master | i-0c6dbb3e33f8bdd48 | ✓ Running | t2.micro | ✓ 2/2 checks passed | No alarms + | ap-south-1b | ec2-3-108-96- |
| □ | node1 | i-0f4be961a77573a88 | ✓ Running | t2.micro | ✓ 2/2 checks passed | No alarms + | ap-south-1b | ec2-43-204-1 |
| □ | node2 | i-0dec173b71f0517f9 | ✓ Running | t2.micro | ✓ 2/2 checks passed | No alarms + | ap-south-1b | ec2-35-154-1 |



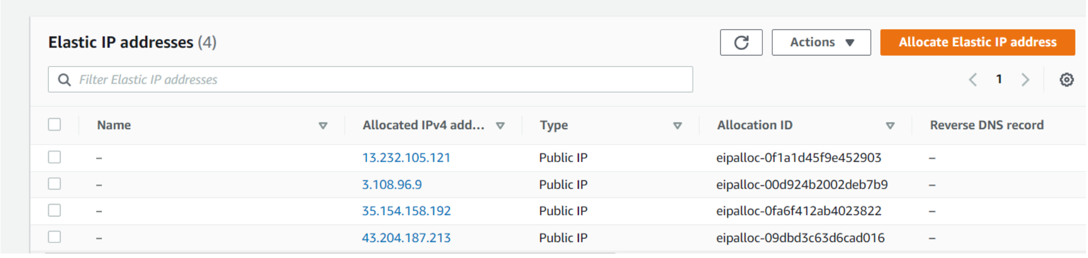
| Public IPv4 DNS | Public IPv4 ... | Elastic IP | IPv6 IPs | Monitoring | Security group name | Key name | Launch time |
|---|---|---|---|---|---|---|---|
| ec2-13-232-105-121.ap... | 13.232.105.121 | 13.232.105.121 | – | disabled | ec2_k8_group | aws_ec2 | 2022/05/31 02 |
| ec2-3-108-96-9.ap-sout... | 3.108.96.9 | 3.108.96.9 | – | disabled | ec2_k8_group | aws_ec2 | 2022/05/31 02 |
| ec2-43-204-187-213.ap... | 43.204.187.213 | 43.204.187.213 | – | disabled | ec2_k8_group | aws_ec2 | 2022/05/31 02 |
| ec2-35-154-158-192.ap... | 35.154.158.192 | 35.154.158.192 | – | disabled | ec2_k8_group | aws_ec2 | 2022/05/31 02 |

### b) Security Group



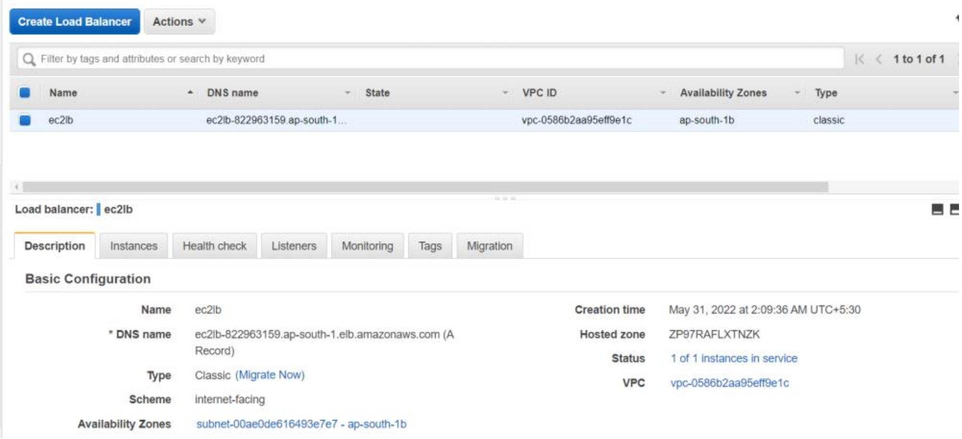
| □ | Name | Security group ID | Security group name | VPC ID | Description | Owner |
|---|---|---|---|---|---|---|
| □ | – | sg-098e3b87b1b332327 | default | vpc-0586b2aa95eff9e1c | default VPC security gr... | 319921805203 |

### c) Elastic IP Address



| □ | Name | Allocated IPv4 add... | Type | Allocation ID | Reverse DNS record |
|---|---|---|---|---|---|
| □ | – | 13.232.105.121 | Public IP | eipalloc-0f1a1d45f9e452903 | – |
| □ | – | 3.108.96.9 | Public IP | eipalloc-00d924b2002deb7b9 | – |
| □ | – | 35.154.158.192 | Public IP | eipalloc-0fa6f412ab4023822 | – |
| □ | – | 43.204.187.213 | Public IP | eipalloc-09dbd3c63d6cad016 | – |

d) Load Balancer



| Name | DNS name | State | VPC ID | Availability Zones | Type |
|---|---|---|---|---|---|
| ec2lb | ec2lb-822963159.ap-south-1... | | vpc-0586b2aa95eff9e1c | ap-south-1b | classic |

Load balancer: ec2lb

Basic Configuration

| | | | |
|---|---|---|---|
| Name | ec2lb | Creation time | May 31, 2022 at 2:09:36 AM UTC+5:30 |
| * DNS name | ec2lb-822963159.ap-south-1.elb.amazonaws.com (A Record) | Hosted zone | ZP97RAFLXTNZK |
| | | Status | 1 of 1 instances in service |
| Type | Classic (Migrate Now) | VPC | vpc-0586b2aa95eff9e1c |
| Scheme | internet-facing | | |
| Availability Zones | subnet-00ae0de616493e7e7 - ap-south-1b | | |

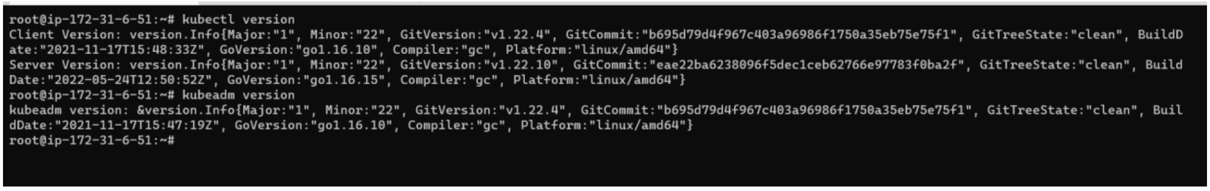
8. SSH into Master Node and Verify Deployed Objects

a) SSH into Master Node

Command: ssh -i aws_ec2.pem ubuntu@3.108.96.9

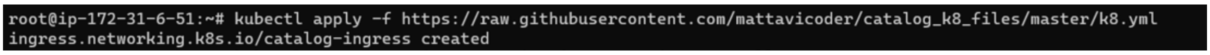
b) Verify K8 Version

Command: **kubectl version, kubeadm version**



```
root@ip-172-31-6-51:~# kubectl version
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.4", GitCommit:"b695d79d4f967c403a96986f1750a35eb75e75f1", GitTreeState:"clean", BuildD
ate:"2021-11-17T15:48:33Z", GoVersion:"go1.16.10", Compiler:"gc", Platform:"linux/amd64"}
Server Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.10", GitCommit:"eae22ba6238096f5dec1ceb62766e97783f0ba2f", GitTreeState:"clean", Build
Date:"2022-05-24T12:50:52Z", GoVersion:"go1.16.15", Compiler:"gc", Platform:"linux/amd64"}
root@ip-172-31-6-51:~# kubeadm version
kubeadm version: &version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.4", GitCommit:"b695d79d4f967c403a96986f1750a35eb75e75f1", GitTreeState:"clean", Buil
dDate:"2021-11-17T15:47:19Z", GoVersion:"go1.16.10", Compiler:"gc", Platform:"linux/amd64"}
root@ip-172-31-6-51:~#
```

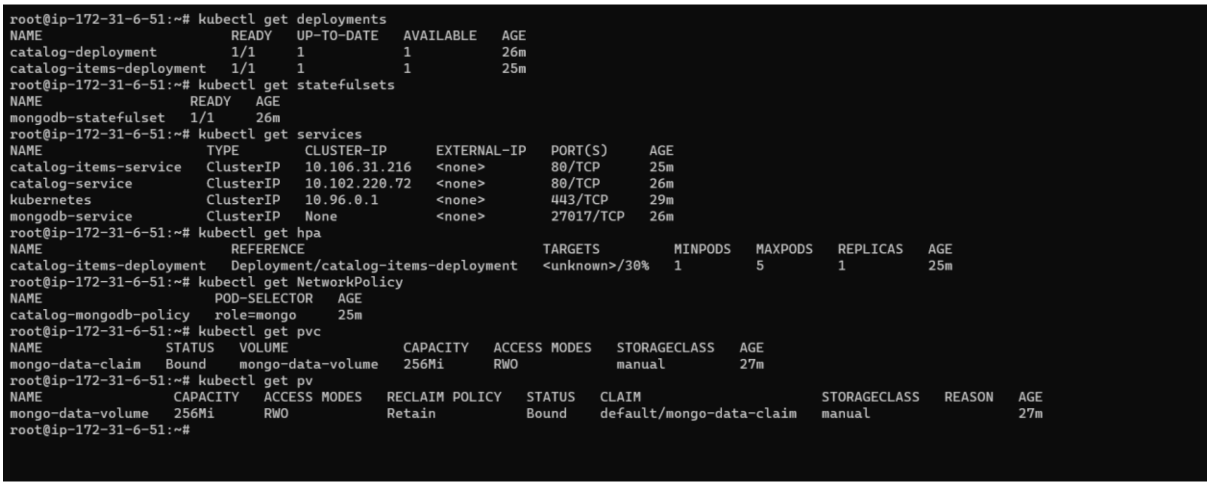
c) Running Ingress services for Deployed Objects



```
root@ip-172-31-6-51:~# kubectl apply -f https://raw.githubusercontent.com/mattavicoder/catalog_k8_files/master/k8.yml
ingress.networking.k8s.io/catalog-ingress created
```

**Note:** Unable to include this in automation process (Pods Role) because, Deployment Objects status needs to be ready before executing this

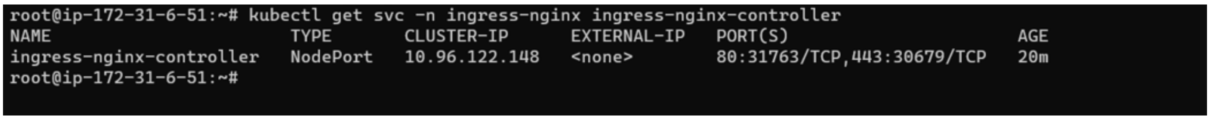
d) Verify K8 Objects



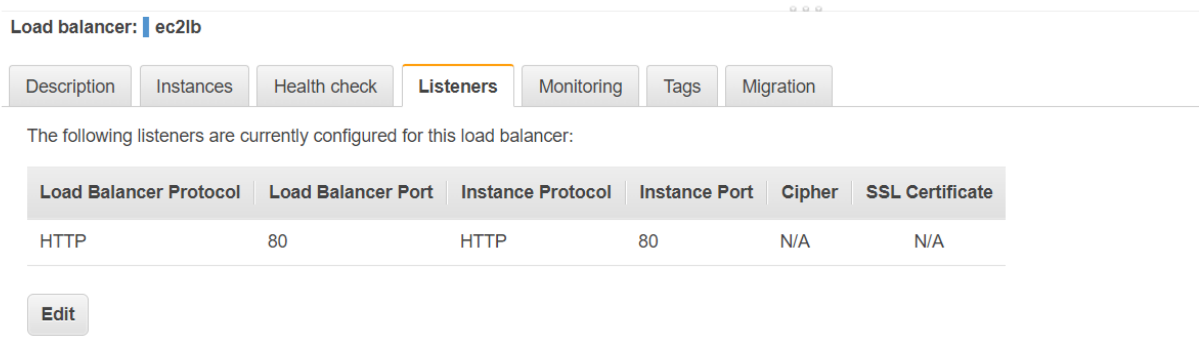
```
root@ip-172-31-6-51:~# kubectl get deployments
NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
catalog-deployment         1/1     1            1           26m
catalog-items-deployment   1/1     1            1           25m
root@ip-172-31-6-51:~# kubectl get statefulsets
NAME                  READY   AGE
mongodb-statefulset   1/1     26m
root@ip-172-31-6-51:~# kubectl get services
NAME                    TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)     AGE
catalog-items-service   ClusterIP   10.106.31.216   <none>        80/TCP      25m
catalog-service         ClusterIP   10.102.220.72   <none>        80/TCP      26m
kubernetes              ClusterIP   10.96.0.1       <none>        443/TCP     29m
mongodb-service         ClusterIP   None            <none>        27017/TCP   26m
root@ip-172-31-6-51:~# kubectl get hpa
NAME                       REFERENCE                             TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
catalog-items-deployment   Deployment/catalog-items-deployment   <unknown>/30%    1         5         1          25m
root@ip-172-31-6-51:~# kubectl get NetworkPolicy
NAME                   POD-SELECTOR   AGE
catalog-mongodb-policy   role=mongo     25m
root@ip-172-31-6-51:~# kubectl get pvc
NAME                STATUS   VOLUME              CAPACITY   ACCESS MODES   STORAGECLASS   AGE
mongo-data-claim    Bound    mongo-data-volume   256Mi      RWO            manual         27m
root@ip-172-31-6-51:~# kubectl get pv
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                      STORAGECLASS   REASON   AGE
mongo-data-volume   256Mi      RWO            Retain           Bound    default/mongo-data-claim   manual                  27m
root@ip-172-31-6-51:~#
```

e) Mapping Ingress Nginx Controller Port to AWS Load Balancer Instance Port
Command: **kubectl get svc -n ingress-nginx ingress-nginx-controller**

Mapping 31763 port to AWS load balancer 80 port



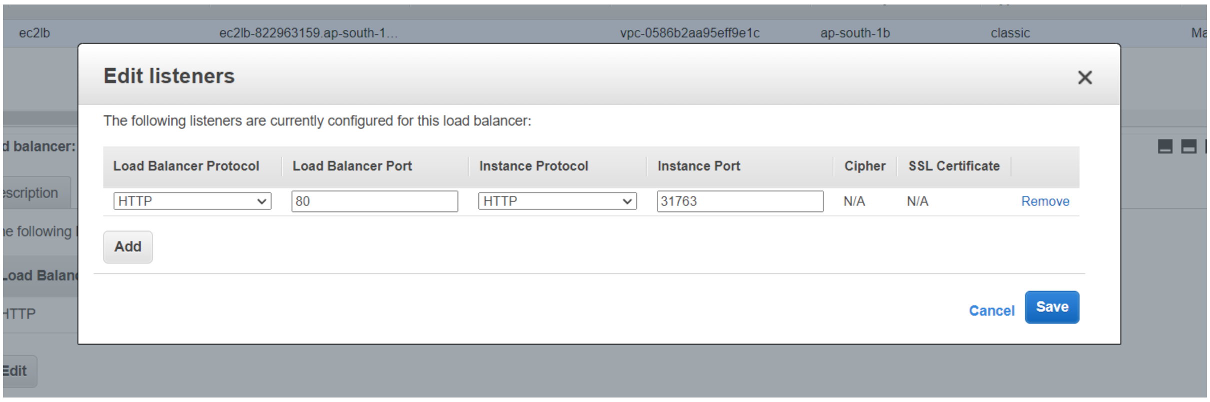
```
root@ip-172-31-6-51:~# kubectl get svc -n ingress-nginx ingress-nginx-controller
NAME                       TYPE       CLUSTER-IP     EXTERNAL-IP   PORT(S)                      AGE
ingress-nginx-controller   NodePort   10.96.122.148  <none>        80:31763/TCP,443:30679/TCP   20m
root@ip-172-31-6-51:~#
```



Load balancer: ec2lb

The following listeners are currently configured for this load balancer:

| Load Balancer Protocol | Load Balancer Port | Instance Protocol | Instance Port | Cipher | SSL Certificate |
|---|---|---|---|---|---|
| HTTP | 80 | HTTP | 80 | N/A | N/A |



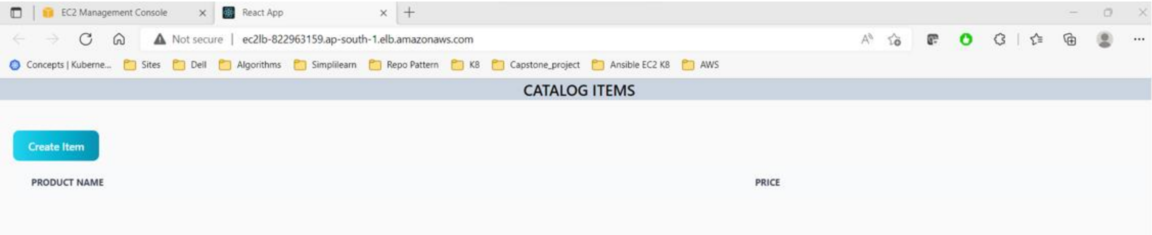
ec2lb ec2lb-822963159.ap-south-1... vpc-0586b2aa95eff9e1c ap-south-1b classic

Edit listeners

The following listeners are currently configured for this load balancer:

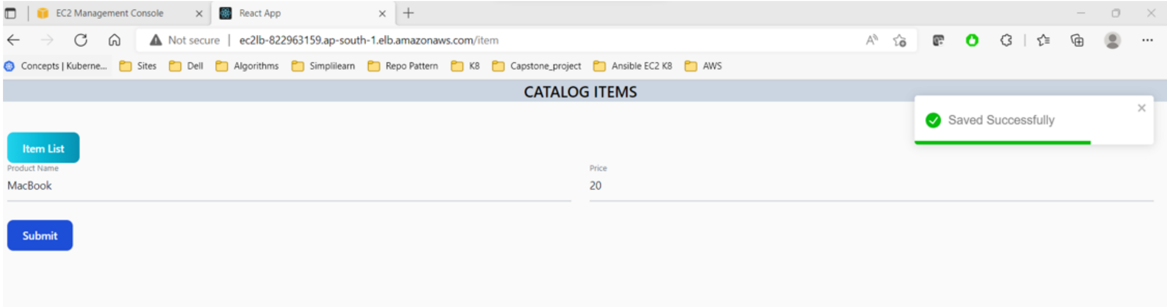
| Load Balancer Protocol | Load Balancer Port | Instance Protocol | Instance Port | Cipher | SSL Certificate | |
|---|---|---|---|---|---|---|
| HTTP | 80 | HTTP | 31763 | N/A | N/A | Remove |

9. Application URL Access ( http://ec2lb-822963159.ap-south-1.elb.amazonaws.com )



CATALOG ITEMS

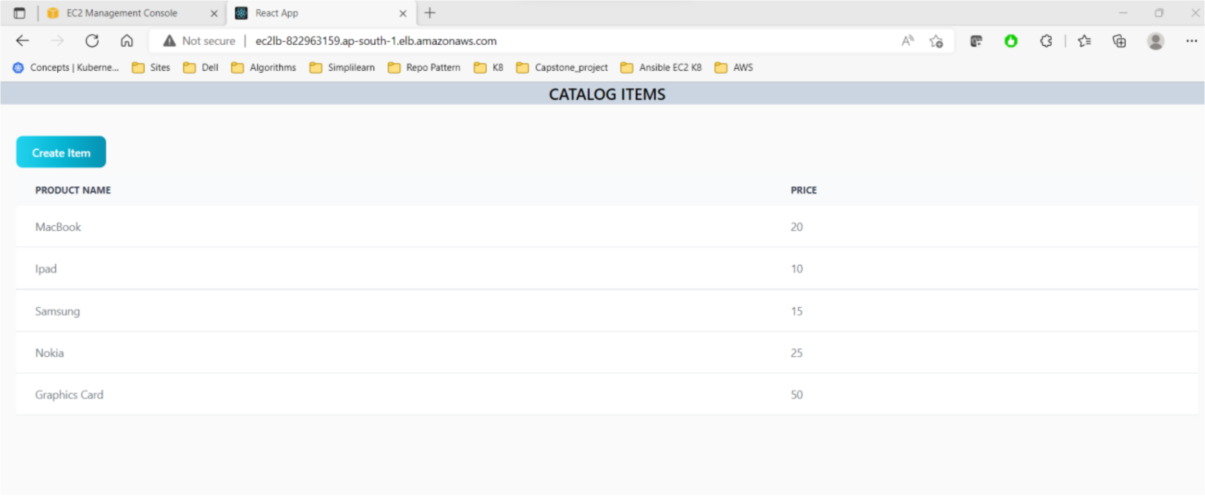
| PRODUCT NAME | PRICE |
|---|---|



CATALOG ITEMS

Saved Successfully

Product Name: MacBook
Price: 20



CATALOG ITEMS

| PRODUCT NAME | PRICE |
|---|---|
| MacBook | 20 |
| Ipad | 10 |
| Samsung | 15 |
| Nokia | 25 |
| Graphics Card | 50 |

10. Creating User ‘areddy’ and Assigning permissions

a) Creating user directory under ‘/home’ folder

Command: **mkdir /home/areddy**

Command: **cd /home/areddy**

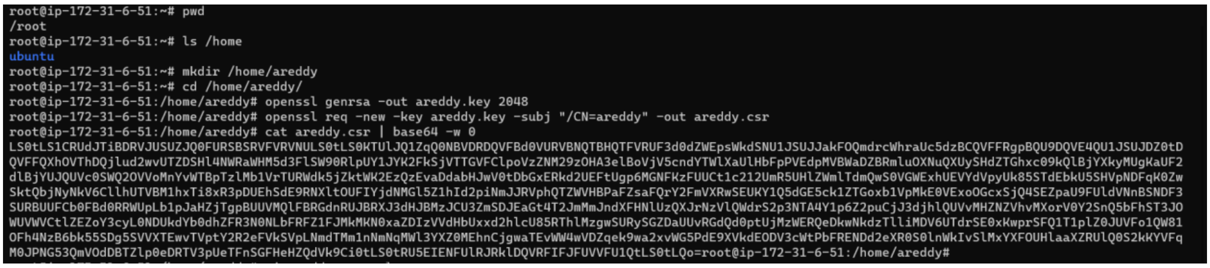
b) Generating a private key for areddy user areddy.key

Command: **openssl genrsa -out ~/areddy/areddy.key 2048**

c) Creating csr request

Command: **openssl req -new -key ~/areddy/areddy.key -subj "/CN=areddy" -out ~/areddy/areddy.csr**

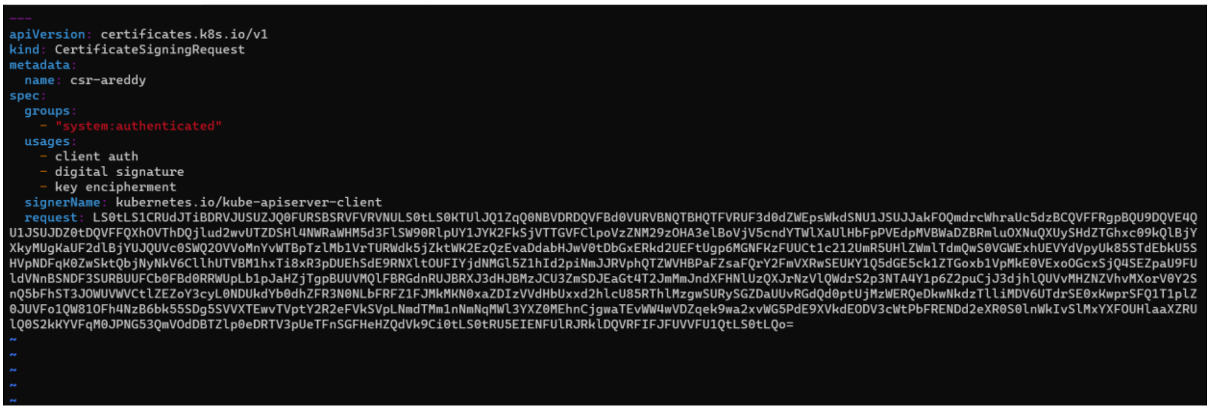
Command: cat ~/areddy/areddy.csr | base64 -w 0



```
root@ip-172-31-6-51:~# pwd
/root
root@ip-172-31-6-51:~# ls /home
ubuntu
root@ip-172-31-6-51:~# mkdir /home/areddy
root@ip-172-31-6-51:~# cd /home/areddy/
root@ip-172-31-6-51:/home/areddy# openssl genrsa -out areddy.key 2048
root@ip-172-31-6-51:/home/areddy# openssl req -new -key areddy.key -subj "/CN=areddy" -out areddy.csr
root@ip-172-31-6-51:/home/areddy# cat areddy.csr | base64 -w 0
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSBSRVFVRVNULS0tLS0KTUlJQ1ZqQ0NBVDRDQVFBd0VURVBNQTBHQTFVRUF3d0dZWEpsWkdSNU1JSUJJakFOQmdrcWhraUc5dzBCQVFFRgpBQU9DQVE4QU1JSUJDZ0tD
QVFFQXhOVThDQjlud2wvUTZDSHl4NWRaWHM5d3FlSW90RlpUY1JYK2FkSjVTTGVFClpoVzZNM29zOHA3elBoVjV5cndYTWlXaUlHbFpPVEdpMVBWaDZBRmluOXNuQXUySHdZTGhxc09kQlBjYXkyMUgKaUF2
dlBjYUJQUVc0SWQ2OVVoMnYvWTBpTzlMb1VrTURWdk5jZktWK2EzQzEvaDdabHJwV0tDbGxERkd2UEFtUgp6MGNFKzFUUCt1c212UmR5UHlZWmlTdmQwS0VGWExhUEVYdVpyUk85STdEbkU5SHVpNDFqK0Zw
SktQbjNyNkV6CllhUTVBM1hxTi8xR3pDUEhSdE9RNXltOUFIYjdNMGl5Z1hId2piNmJJRVphQTZWVHBPaFZsaFQrY2FmVXRwSEUKY1Q5dGE5ck1ZTGoxb1VpMkE0VExoOGcxSjQ4SEZpaU9FUldVNnBSNDF3
SURBUUFCb0FBd0RRWUpLb1pJaHZjTgpBUUVMQlFBRGdnRUJBRXJ3dHJBMzJCU3ZmSDJEaGt4T2JmMmJndXFHNlUzQXJrNzVlQWdrS2p3NTA4Y1p6Z2puCjJ3djhlQUVvMHZNZVhvMXorV0Y2SnQ5bFhST3JO
WUVWVCtlZEZoY3cyL0NDUkdYb0dhZFR3N0NLbFRFZ1FJMkMKN0xaZDIzVVdHbUxxd2hlcU85RThlMzgwSURySGZDaUUvRGdQd0ptUjMzWERReDkwNkdzTlliMDV6UTdrSE0xKwprSFQ1T1plZ0JUVFo1QW81
OFh4NzB6bk55SDg5SVVXTEwvTVptY2R2eFVkSVpLNmdTMm1nNmNqMWl3YXZ0MEhnCjgwaTEvWW4wVDZqek9wa2xvWG5PdE9XVkdEODV3cWtPbFRENDd2eXR0S0lnWkIvSlMxYXFOUHlaaXZRUlQ0S2kKYVFq
M0JPNG53QmVOdDBTZlp0eDRTV3pUeTFnSGFHeHZQdVk9Ci0tLS0tRU5EIENFUlRJRklDQVRFIFJFUVVFU1QtLS0tLQo=root@ip-172-31-6-51:/home/areddy#
```

d) Certificate Signing Request file areddy_csr.yml

Command: **kubectl apply -f areddy_csr.yml**



```
---
apiVersion: certificates.k8s.io/v1
kind: CertificateSigningRequest
metadata:
  name: csr-areddy
spec:
  groups:
    - "system:authenticated"
  usages:
    - client auth
    - digital signature
    - key encipherment
  signerName: kubernetes.io/kube-apiserver-client
  request: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSBSRVFVRVNULS0tLS0KTUlJQ1ZqQ0NBVDRDQVFBd0VURVBNQTBHQTFVRUF3d0dZWEpsWkdSNU1JSUJJakFOQmdrcWhraUc5dzBCQVFFRgpBQU9DQVE4Q
U1JSUJDZ0tDQVFFQXhOVThDQjlud2wvUTZDSHl4NWRaWHM5d3FlSW90RlpUY1JYK2FkSjVTTGVFClpoVzZNM29zOHA3elBoVjV5cndYTWlXaUlHbFpPVEdpMVBWaDZBRmluOXNuQXUySHdZTGhxc09kQlBjY
XkyMUgKaUF2dlBjYUJQUVc0SWQ2OVVoMnYvWTBpTzlMb1VrTURWdk5jZktWK2EzQzEvaDdabHJwV0tDbGxERkd2UEFtUgp6MGNFKzFUUCt1c212UmR5UHlZWmlTdmQwS0VGWExhUEVYdVpyUk85STdEbkU5S
HVpNDFqK0ZwSktQbjNyNkV6CllhUTVBM1hxTi8xR3pDUEhSdE9RNXltOUFIYjdNMGl5Z1hId2piNmJJRVphQTZWVHBPaFZsaFQrY2FmVXRwSEUKY1Q5dGE5ck1ZTGoxb1VpMkE0VExoOGcxSjQ4SEZpaU9FU
ldVNnBSNDF3SURBUUFCb0FBd0RRWUpLb1pJaHZjTgpBUUVMQlFBRGdnRUJBRXJ3dHJBMzJCU3ZmSDJEaGt4T2JmMmJndXFHNlUzQXJrNzVlQWdrS2p3NTA4Y1p6Z2puCjJ3djhlQUVvMHZNZVhvMXorV0Y2S
nQ5bFhST3JOWUVWVCtlZEZoY3cyL0NDUkdYb0dhZFR3N0NLbFRFZ1FJMkMKN0xaZDIzVVdHbUxxd2hlcU85RThlMzgwSURySGZDaUUvRGdQd0ptUjMzWERQeDkwNkdzTlliMDV6UTdrSE0xKwprSFQ1T1plZ
0JUVFo1QW81OFh4NzB6bk55SDg5SVVXTEwvTVptY2R2eFVkSVpLNmdTMm1nNmNqMWl3YXZ0MEhnCjgwaTEvWW4wVDZqek9wa2xvWG5PdE9XVkdEODV3cWtPbFRENDd2eXR0S0lnWkIvSlMxYXFOUHlaaXZRU
lQ0S2kKYVFqM0JPNG53QmVOdDBTZlp0eDRTV3pUeTFnSGFHeHZQdVk9Ci0tLS0tRU5EIENFUlRJRklDQVRFIFJFUVVFU1QtLS0tLQo=
~
~
~
~
~
```

e) Miscellaneous commands for User setup

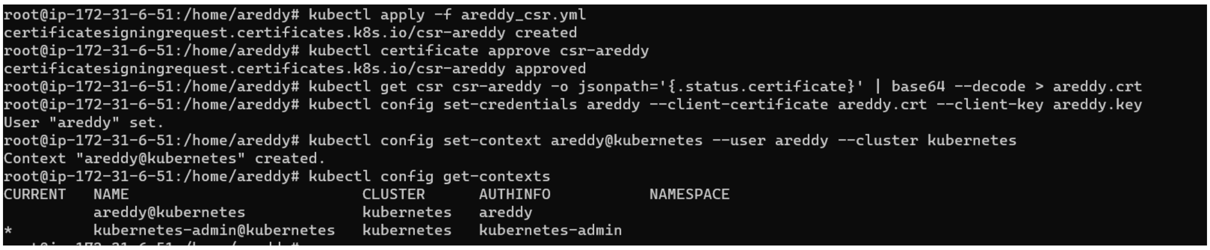
Approve csr-areddy: **kubectl certificate approve csr-areddy**

Save the certificate in a file: **kubectl get csr csr-areddy -o jsonpath='{.status.certificate}' | base64 --decode > ~/areddy/areddy.crt**

Set user in kube config: **kubectl config set-credentials areddy --client-certificate ~/areddy/areddy.crt --client-key ~/areddy/areddy.key**

Create a new context to use areddy user: **kubectl config set-context areddy@kubernetes --user areddy --cluster kubernetes**

Set the new context as default: **kubectl config use-context areddy@kubernetes**



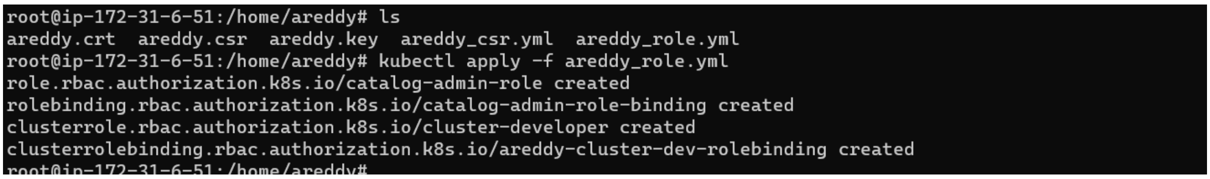
```
root@ip-172-31-6-51:/home/areddy# kubectl apply -f areddy_csr.yml
certificatesigningrequest.certificates.k8s.io/csr-areddy created
root@ip-172-31-6-51:/home/areddy# kubectl certificate approve csr-areddy
certificatesigningrequest.certificates.k8s.io/csr-areddy approved
root@ip-172-31-6-51:/home/areddy# kubectl get csr csr-areddy -o jsonpath='{.status.certificate}' | base64 --decode > areddy.crt
root@ip-172-31-6-51:/home/areddy# kubectl config set-credentials areddy --client-certificate areddy.crt --client-key areddy.key
User "areddy" set.
root@ip-172-31-6-51:/home/areddy# kubectl config set-context areddy@kubernetes --user areddy --cluster kubernetes
Context "areddy@kubernetes" created.
root@ip-172-31-6-51:/home/areddy# kubectl config get-contexts
CURRENT   NAME                          CLUSTER      AUTHINFO           NAMESPACE
          areddy@kubernetes             kubernetes   areddy
*         kubernetes-admin@kubernetes   kubernetes   kubernetes-admin
```

f) Create Role, Role Binding, Cluster Role, Cluster Role Binding for 'areddy' user

Created a file areddy_role.yml file using the source ( catalog_k8_files/roles.yaml at main · mattavicoder/catalog_k8_files (github.com))

```yaml
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  namespace: default
  name: catalog-admin-role
rules:
  - apiGroups: [""]
    resources:
      [
        "services",
        "endpoints",
        "pods",
        "persistentvolumeclaims",
        "persistentvolumes",
        "quota",
      ]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]
  - apiGroups: ["apps"]
    resources: ["deployments", "statefulsets"]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]
  - apiGroups: ["extensions"]
    resources: ["ingresses"]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]
  - apiGroups: ["networking.k8s.io"]
    resources: ["networkpolicies", "ingressclasses", "ingresses"]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]
  - apiGroups: ["storage.k8s.io"]
    resources: ["storageclasses"]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]
  - apiGroups: ["autoscaling"]
    resources: ["horizontalpodautoscalers"]
    verbs: ["get", "list", "watch", "create", "update", "patch", "delete"]



---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: cluster-developer
rules:
  - apiGroups: [""]
    resources: ["nodes", "persistentvolumeclaims", "persistentvolumes"]
    verbs: ["get", "list", "create"]
  - apiGroups: [""]
    resources: ["namespaces"]
    verbs: ["get", "list", "create"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: areddy-cluster-dev-rolebinding
subjects:
  - kind: User
    name: areddy
    apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: ClusterRole
  name: cluster-developer
  apiGroup: rbac.authorization.k8s.io
```

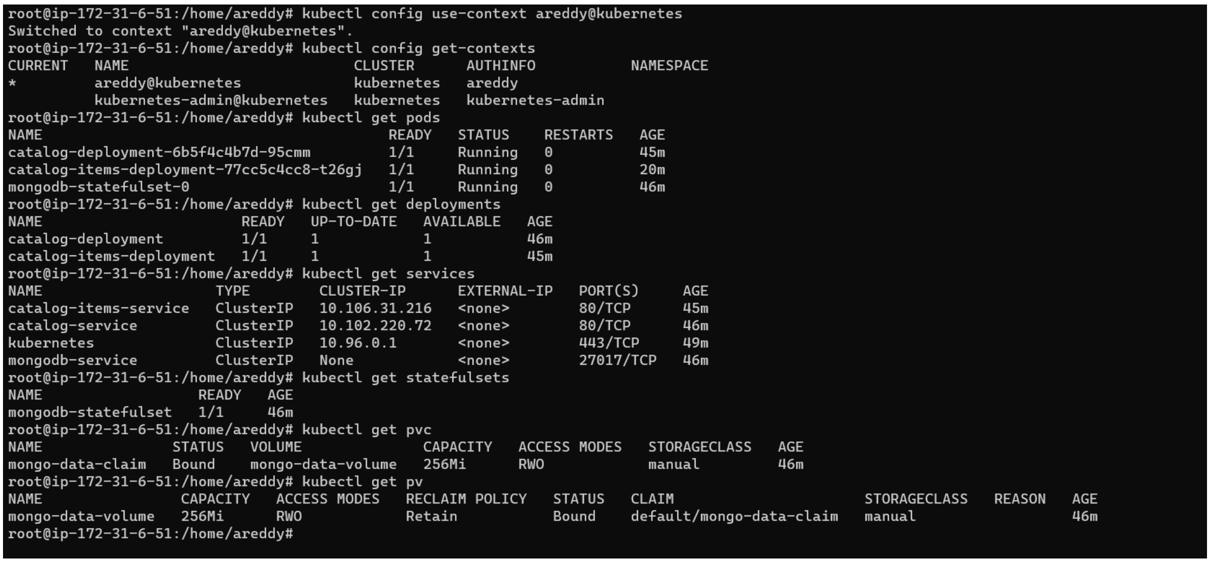
Apply roles to ‘areddy’ user: **kubectl apply -f areddy_role.yml**



```
root@ip-172-31-6-51:/home/areddy# ls
areddy.crt  areddy.csr  areddy.key  areddy_csr.yml  areddy_role.yml
root@ip-172-31-6-51:/home/areddy# kubectl apply -f areddy_role.yml
role.rbac.authorization.k8s.io/catalog-admin-role created
rolebinding.rbac.authorization.k8s.io/catalog-admin-role-binding created
clusterrole.rbac.authorization.k8s.io/cluster-developer created
clusterrolebinding.rbac.authorization.k8s.io/areddy-cluster-dev-rolebinding created
root@ip-172-31-6-51:/home/areddy#
```

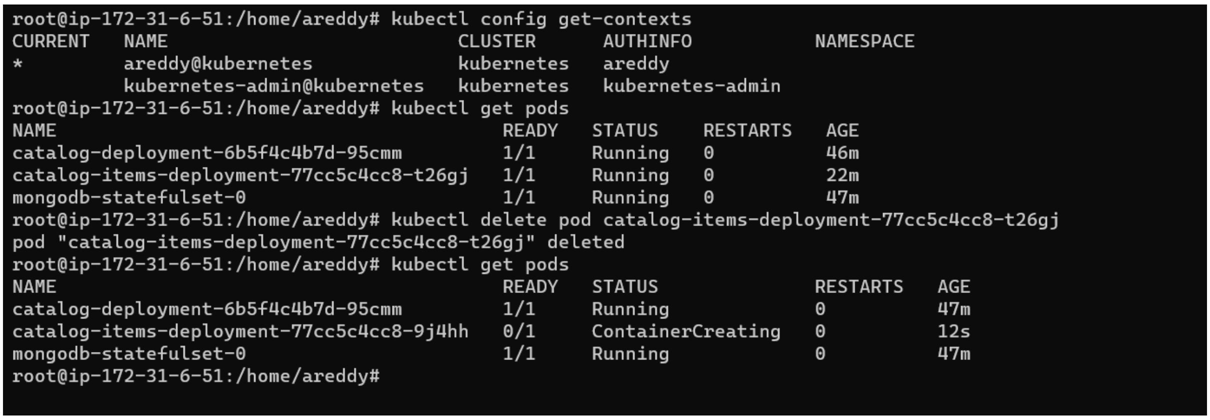
g) Verify the Role permission

Get permissions



```
root@ip-172-31-6-51:/home/areddy# kubectl config use-context areddy@kubernetes
Switched to context "areddy@kubernetes".
root@ip-172-31-6-51:/home/areddy# kubectl config get-contexts
CURRENT   NAME                          CLUSTER      AUTHINFO           NAMESPACE
*         areddy@kubernetes             kubernetes   areddy
          kubernetes-admin@kubernetes   kubernetes   kubernetes-admin
root@ip-172-31-6-51:/home/areddy# kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm      1/1     Running   0          45m
catalog-items-deployment-77cc5c4cc8-t26gj   1/1     Running   0          20m
mongodb-statefulset-0                    1/1     Running   0          46m
root@ip-172-31-6-51:/home/areddy# kubectl get deployments
NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
catalog-deployment         1/1     1            1           46m
catalog-items-deployment   1/1     1            1           45m
root@ip-172-31-6-51:/home/areddy# kubectl get services
NAME                    TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)     AGE
catalog-items-service   ClusterIP   10.106.31.216   <none>        80/TCP      45m
catalog-service         ClusterIP   10.102.220.72   <none>        80/TCP      46m
kubernetes              ClusterIP   10.96.0.1       <none>        443/TCP     49m
mongodb-service         ClusterIP   None            <none>        27017/TCP   46m
root@ip-172-31-6-51:/home/areddy# kubectl get statefulsets
NAME                  READY   AGE
mongodb-statefulset   1/1     46m
root@ip-172-31-6-51:/home/areddy# kubectl get pvc
NAME                STATUS   VOLUME              CAPACITY   ACCESS MODES   STORAGECLASS   AGE
mongo-data-claim    Bound    mongo-data-volume   256Mi      RWO            manual         46m
root@ip-172-31-6-51:/home/areddy# kubectl get pv
NAME                CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                      STORAGECLASS   REASON   AGE
mongo-data-volume   256Mi      RWO            Retain           Bound    default/mongo-data-claim   manual                  46m
root@ip-172-31-6-51:/home/areddy#
```

Delete Permissions



```
root@ip-172-31-6-51:/home/areddy# kubectl config get-contexts
CURRENT   NAME                          CLUSTER      AUTHINFO           NAMESPACE
*         areddy@kubernetes             kubernetes   areddy
          kubernetes-admin@kubernetes   kubernetes   kubernetes-admin
root@ip-172-31-6-51:/home/areddy# kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm         1/1     Running   0          46m
catalog-items-deployment-77cc5c4cc8-t26gj   1/1     Running   0          22m
mongodb-statefulset-0                       1/1     Running   0          47m
root@ip-172-31-6-51:/home/areddy# kubectl delete pod catalog-items-deployment-77cc5c4cc8-t26gj
pod "catalog-items-deployment-77cc5c4cc8-t26gj" deleted
root@ip-172-31-6-51:/home/areddy# kubectl get pods
NAME                                        READY   STATUS              RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm         1/1     Running             0          47m
catalog-items-deployment-77cc5c4cc8-9j4hh   0/1     ContainerCreating   0          12s
mongodb-statefulset-0                       1/1     Running             0          47m
root@ip-172-31-6-51:/home/areddy#
```

11. ECTD Database Snapshot

a) Install ETCD

Execute the commands in the following order

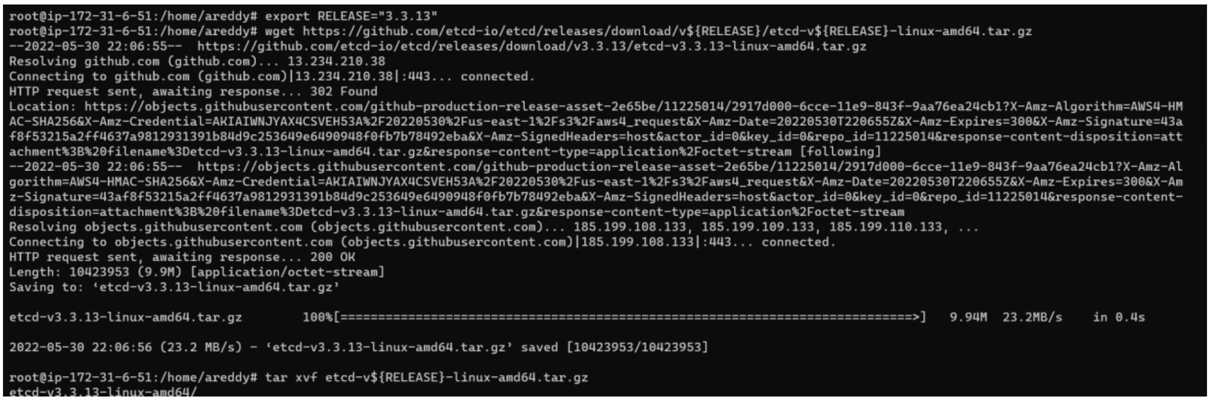
**export RELEASE="3.3.13"**

**wget https://github.com/etcd-io/etcd/releases/download/v${RELEASE}/etcd-v${RELEASE}-linux-amd64.tar.gz**

**tar xvf etcd-v${RELEASE}-linux-amd64.tar.gz**

**cd etcd-v${RELEASE}-linux-amd64**

**mv etcdctl /usr/local/bin**



```
root@ip-172-31-6-51:/home/areddy# export RELEASE="3.3.13"
root@ip-172-31-6-51:/home/areddy# wget https://github.com/etcd-io/etcd/releases/download/v${RELEASE}/etcd-v${RELEASE}-linux-amd64.tar.gz
--2022-05-30 22:06:55--  https://github.com/etcd-io/etcd/releases/download/v3.3.13/etcd-v3.3.13-linux-amd64.tar.gz
Resolving github.com (github.com)... 13.234.210.38
Connecting to github.com (github.com)|13.234.210.38|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/11225014/2917d000-6cce-11e9-843f-9aa76ea24cb1?X-Amz-Algorithm=AWS4-HM
AC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220530%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220530T220655Z&X-Amz-Expires=300&X-Amz-Signature=43a
f8f53215a2ff4637a9812931391b84d9c253649e6490948f0fb7b78492eba&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=11225014&response-content-disposition=att
achment%3B%20filename%3Detcd-v3.3.13-linux-amd64.tar.gz&response-content-type=application%2Foctet-stream [following]
--2022-05-30 22:06:55--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/11225014/2917d000-6cce-11e9-843f-9aa76ea24cb1?X-Amz-Al
gorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220530%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20220530T220655Z&X-Amz-Expires=300&X-Am
z-Signature=43af8f53215a2ff4637a9812931391b84d9c253649e6490948f0fb7b78492eba&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=11225014&response-content-
disposition=attachment%3B%20filename%3Detcd-v3.3.13-linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10423953 (9.9M) [application/octet-stream]
Saving to: ‘etcd-v3.3.13-linux-amd64.tar.gz’

etcd-v3.3.13-linux-amd64.tar.gz        100%[==========================================================================>]   9.94M  23.2MB/s    in 0.4s

2022-05-30 22:06:56 (23.2 MB/s) - ‘etcd-v3.3.13-linux-amd64.tar.gz’ saved [10423953/10423953]

root@ip-172-31-6-51:/home/areddy# tar xvf etcd-v${RELEASE}-linux-amd64.tar.gz
etcd-v3.3.13-linux-amd64/
```

b) Backup ETCD database

Create a directory for backup location

Command: **mkdir etcd-catalog-backup**

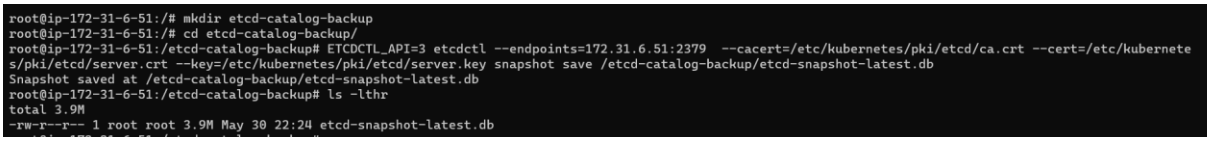
**cd etcd-catalog-backup**

Using the IP and Port number of ETCD 172.31.6.51:2379 (using ‘ps -ef| grep etcd’ )

Saving the ETCD Snapshot as ‘etcd-snapshot-latest.db’

Command: **ETCDCTL_API=3 etcdctl --endpoints=172.31.6.51:2379 --cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernetes/pki/etcd/server.crt --key=/etc/kubernetes/pki/etcd/server.key snapshot save /etcd-catalog-backup/etcd-snapshot-latest.db**

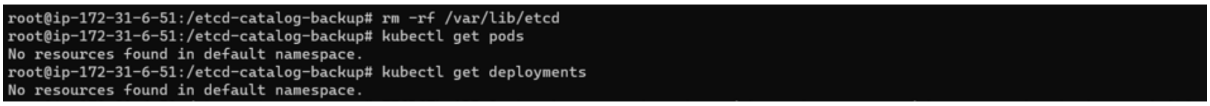
Verify DB file: **ls -lthr**



```
root@ip-172-31-6-51:/# mkdir etcd-catalog-backup
root@ip-172-31-6-51:/# cd etcd-catalog-backup/
root@ip-172-31-6-51:/etcd-catalog-backup# ETCDCTL_API=3 etcdctl --endpoints=172.31.6.51:2379  --cacert=/etc/kubernetes/pki/etcd/ca.crt --cert=/etc/kubernete
s/pki/etcd/server.crt --key=/etc/kubernetes/pki/etcd/server.key snapshot save /etcd-catalog-backup/etcd-snapshot-latest.db
Snapshot saved at /etcd-catalog-backup/etcd-snapshot-latest.db
root@ip-172-31-6-51:/etcd-catalog-backup# ls -lthr
total 3.9M
-rw-r--r-- 1 root root 3.9M May 30 22:24 etcd-snapshot-latest.db
```

c) Remove the /var/lib/etcd folder and verify pods, deployments

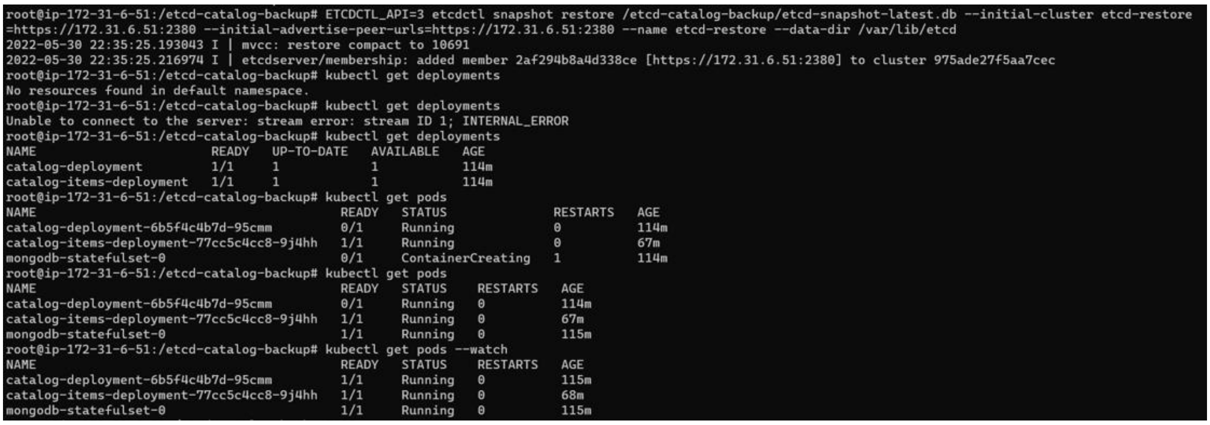
Command: **rm -rf /var/lib/etcd**



```
root@ip-172-31-6-51:/etcd-catalog-backup# rm -rf /var/lib/etcd
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get pods
No resources found in default namespace.
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get deployments
No resources found in default namespace.
```

d) Restore ETCD Snapshot backup and verify pods, deployments

Command: **ETCDCTL_API=3 etcdctl snapshot restore /etcd-catalog-backup/etcd-snapshot-latest.db --initial-cluster etcd-restore=https://172.31.6.51:2380 --initial-advertise-peer-urls=https://172.31.6.51:2380 --name etcd-restore --data-dir /var/lib/etcd**



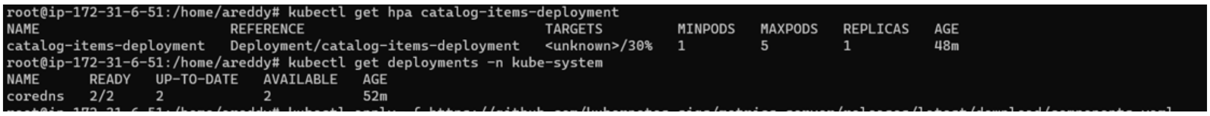
```
root@ip-172-31-6-51:/etcd-catalog-backup# ETCDCTL_API=3 etcdctl snapshot restore /etcd-catalog-backup/etcd-snapshot-latest.db --initial-cluster etcd-restore
=https://172.31.6.51:2380 --initial-advertise-peer-urls=https://172.31.6.51:2380 --name etcd-restore --data-dir /var/lib/etcd
2022-05-30 22:35:25.193043 I | mvcc: restore compact to 10691
2022-05-30 22:35:25.216974 I | etcdserver/membership: added member 2af294b8a4d338ce [https://172.31.6.51:2380] to cluster 975ade27f5aa7cec
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get deployments
No resources found in default namespace.
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get deployments
Unable to connect to the server: stream error: stream ID 1; INTERNAL_ERROR
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get deployments
NAME                      READY   UP-TO-DATE   AVAILABLE   AGE
catalog-deployment        1/1     1            1           114m
catalog-items-deployment  1/1     1            1           114m
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get pods
NAME                                       READY   STATUS              RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm        0/1     Running             0          114m
catalog-items-deployment-77cc5c4cc8-9j4hh  1/1     Running             0          67m
mongodb-statefulset-0                      0/1     ContainerCreating   1          114m
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get pods
NAME                                       READY   STATUS    RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm        0/1     Running   0          114m
catalog-items-deployment-77cc5c4cc8-9j4hh  1/1     Running   0          67m
mongodb-statefulset-0                      1/1     Running   0          115m
root@ip-172-31-6-51:/etcd-catalog-backup# kubectl get pods --watch
NAME                                       READY   STATUS    RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm        1/1     Running   0          115m
catalog-items-deployment-77cc5c4cc8-9j4hh  1/1     Running   0          68m
mongodb-statefulset-0                      1/1     Running   0          115m
```

12. UI configuration setup to Horizontal Scale up on CPU memory goes beyond 30%

a) Verify HorizontalPodAutoscaler has been deployed under Pods Role

Command: **kubectl get hpa catalog-items-deployment**

```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: catalog-items-deployment
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: catalog-items-deployment
  minReplicas: 1
  maxReplicas: 5
  targetCPUUtilizationPercentage: 30
```



```
root@ip-172-31-6-51:/home/areddy# kubectl get hpa catalog-items-deployment
NAME                       REFERENCE                             TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
catalog-items-deployment   Deployment/catalog-items-deployment   <unknown>/30%    1         5         1          48m
root@ip-172-31-6-51:/home/areddy# kubectl get deployments -n kube-system
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
coredns   2/2     2            2           52m
root@ip-172-31-6-51:/home/areddy# kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml
```

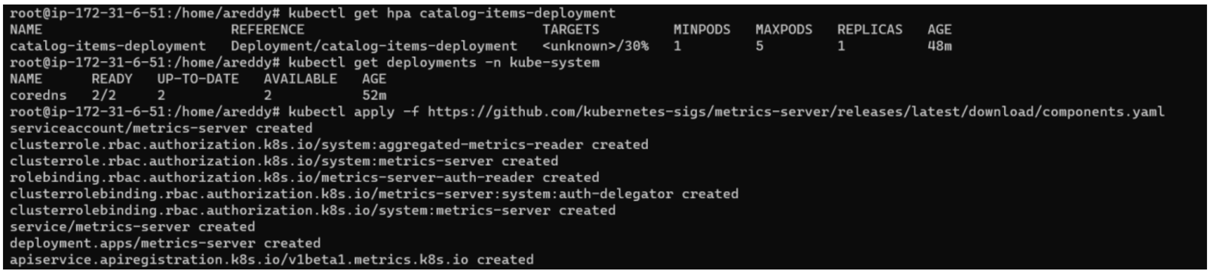
b) Metrics Server Setup

Currently hpa results shows “<unknown>/30%” under Targets (above image)

and under kube-system namespace no metrics-server deployment (above image)

Command: **kubectl apply -f [https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml](https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml)**

Command: **kubectl patch deployment metrics-server -n kube-system --type 'json' -p '[{"op": "add", "path": "/spec/template/spec/containers/0/args/-", "value": "--kubelet-insecure-tls"}]'**

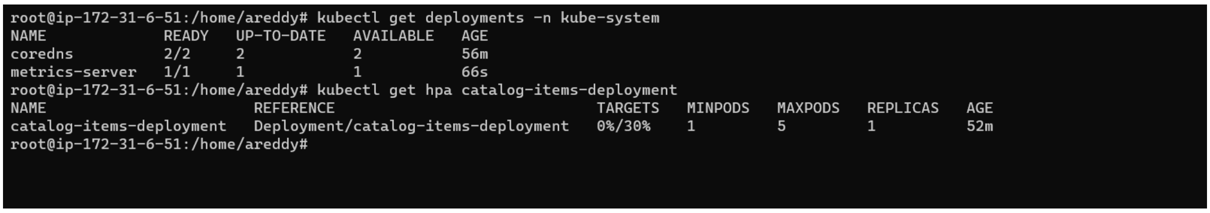


```
root@ip-172-31-6-51:/home/areddy# kubectl get hpa catalog-items-deployment
NAME                       REFERENCE                             TARGETS         MINPODS   MAXPODS   REPLICAS   AGE
catalog-items-deployment   Deployment/catalog-items-deployment   <unknown>/30%   1         5         1          48m
root@ip-172-31-6-51:/home/areddy# kubectl get deployments -n kube-system
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
coredns   2/2     2            2           52m
root@ip-172-31-6-51:/home/areddy# kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml
serviceaccount/metrics-server created
clusterrole.rbac.authorization.k8s.io/system:aggregated-metrics-reader created
clusterrole.rbac.authorization.k8s.io/system:metrics-server created
rolebinding.rbac.authorization.k8s.io/metrics-server-auth-reader created
clusterrolebinding.rbac.authorization.k8s.io/metrics-server:system:auth-delegator created
clusterrolebinding.rbac.authorization.k8s.io/system:metrics-server created
service/metrics-server created
deployment.apps/metrics-server created
apiservice.apiregistration.k8s.io/v1beta1.metrics.k8s.io created
```

Verify the metric server installation

Command**: kubectl get deployments -n kube-system**

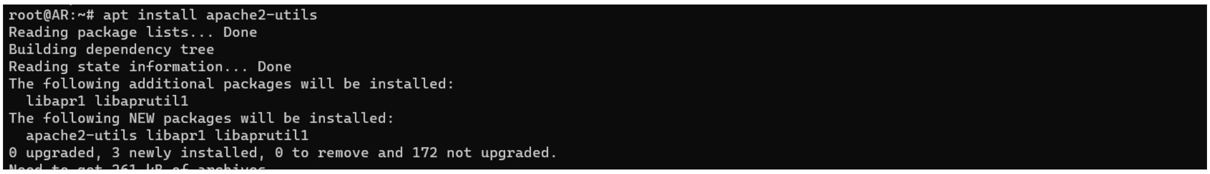
Command: **kubectl get hpa catalog-items-deployment**



```
root@ip-172-31-6-51:/home/areddy# kubectl get deployments -n kube-system
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
coredns          2/2     2            2           56m
metrics-server   1/1     1            1           66s
root@ip-172-31-6-51:/home/areddy# kubectl get hpa catalog-items-deployment
NAME                       REFERENCE                             TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
catalog-items-deployment   Deployment/catalog-items-deployment   0%/30%    1         5         1          52m
root@ip-172-31-6-51:/home/areddy#
```

c) Install apache2-utils

Command: **apt install apache2-utils**



```
root@AR:~# apt install apache2-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libapr1 libaprutil1
The following NEW packages will be installed:
  apache2-utils libapr1 libaprutil1
0 upgraded, 3 newly installed, 0 to remove and 172 not upgraded.
Need to get 261 kB of archives.
```

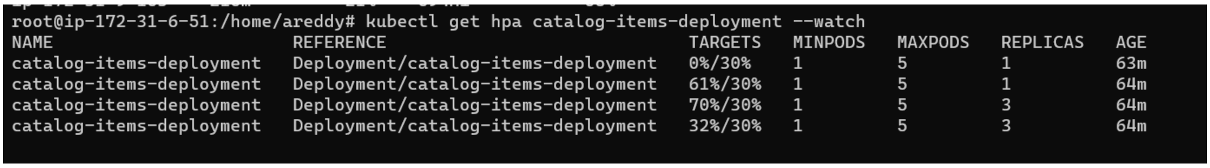
d) Testing the Load to see the behaviour of HorizontalPodAutoScaler on UI deployment

Once CPU utilization reaches 30% new pods should spun up

Command: **ab -n 50000 -c 1000 http://ec2lb-822963159.ap-south-1.elb.amazonaws.com/**

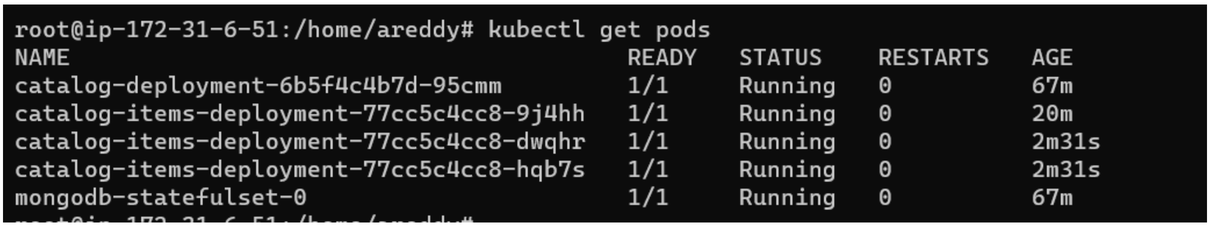
Performs 50000 requests 1000 at a time

Check Load: **kubectl get hpa catalog-items-deployment --watch**



```
root@ip-172-31-6-51:/home/areddy# kubectl get hpa catalog-items-deployment --watch
NAME                      REFERENCE                            TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
catalog-items-deployment  Deployment/catalog-items-deployment  0%/30%    1         5         1          63m
catalog-items-deployment  Deployment/catalog-items-deployment  61%/30%   1         5         1          64m
catalog-items-deployment  Deployment/catalog-items-deployment  70%/30%   1         5         3          64m
catalog-items-deployment  Deployment/catalog-items-deployment  32%/30%   1         5         3          64m
```

Check Pods: **kubectl get pods**



```
root@ip-172-31-6-51:/home/areddy# kubectl get pods
NAME                                      READY   STATUS    RESTARTS   AGE
catalog-deployment-6b5f4c4b7d-95cmm       1/1     Running   0          67m
catalog-items-deployment-77cc5c4cc8-9j4hh 1/1     Running   0          20m
catalog-items-deployment-77cc5c4cc8-dwqhr 1/1     Running   0          2m31s
catalog-items-deployment-77cc5c4cc8-hqb7s 1/1     Running   0          2m31s
mongodb-statefulset-0                     1/1     Running   0          67m
```

Additional 2 pods has been spun up

## Summary

a) **Project and tester details**

A Catalog Items List application using React as FrontEnd, .Net Core API as BackEnd, MongoDb as Database

Developer & Tester: Matta Avinash Reddy

Email: mattavireddy@outlook.com

b) **Links to the Github Repo**

UI Application: https://github.com/mattavicoder/catalog-items

Api Application: https://github.com/mattavicoder/catalog

Ansible Scripts: https://github.com/mattavicoder/aws_ec2_k8_ansible_scripts

K8 Objects yaml files: https://github.com/mattavicoder/catalog_k8_files

Docker Hub UI Image: https://hub.docker.com/r/mattavireddy/catalogitems

Docker Hub API Image: https://hub.docker.com/r/mattavireddy/catalog

Docker Hub MongoDb Image: https://hub.docker.com/_/mongo

c) **Concepts used in the Project**

**Automation Tool:** Ansible

Automation engine tool that automates cloud provisioning, configuration management, application deployment, intra-service orchestration, and many other IT needs.

Leveraging Ansible Roles concept of performing tasks, having its own variables with in a role.

**Cloud:** AWS, where EC2 Ubuntu instances are created using Ansible configuration tool by passing required security crendentials,

Creating Security Groups, Load balancer with Port mapping, Elastic IP Address for EC2 Instances

**Containers & K8:**

Applications are build as Docker images using Docker files.

Using Docker images from Docker Hub, Applications are deployed as Containers on K8 Cluster, Where Horziontal Auto Scale has been configured, on CPU load increases Automatically new pods are spun up, so that load can be decreased

Network Policies has been set between services.

ETCD Database BackUp and Restore snapshots.

Creating user with role bindings

d) **Unique Selling Points**

a. Maintaining Applications always at desired state by Automated configured systems, which leads to consistencies.

b. Automatically deployment and scale the applications based on workload

c. Code version is now at Github, central repository version control.

d. All team can now easily develop and test the application either at cloud or in the system with minimal process.

e. Releases are now faster