In this document we will setup eks cluster using CLI

Install aws cli in linux machine to connect eks cluster.

Install or update latest version of aws cli.

```
[root@developer2-system ~]# curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip_awscliv2.zip
sudo ./aws/install
```

sudo yum remove awscli

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip" unzip awscliv2.zip

sudo ./aws/install

To see the changes exist the login shell and login again

RUN the command aws -version to see the changes.

```
[root@developer2-system =]# aws --version
aws-clt/2:35:59 Python/3:11.8 Linux/0:1.89-99.174.amzn2023.x86_64 exe/x86_64.amzn.2023
[root@developer2-system =]# curl = 0 https://s3.us-west-2.amazonws.com/amazon-eks/1.28.88/2024-04-19/bin/linux/amd64/kubectl
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Received % Xford Average Speed Time Time Time Current
% Total % Total
```

Installing eksctl in linux machine to access or create eks.

Create EKS cluster role.

AWS Dashboard Go to into IAM

Create role

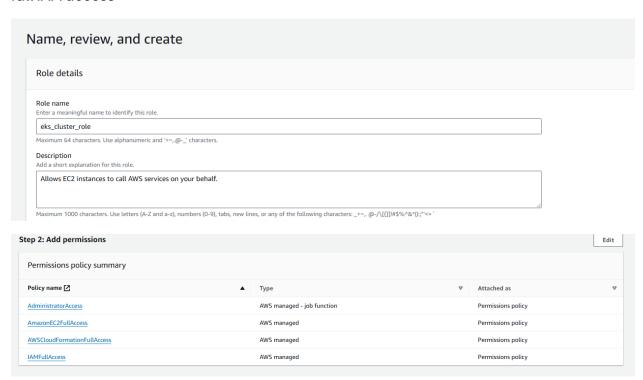
Add permission

fullEC2access

full Administrator

fullCloudformation

full IAM access



Attach ekscluster role k8s_boot_strep linux machine

Update IAM role to linux machine.

Setup kubernetes using eksctl.

Example

eksctl create cluster --name cluster-name \

```
--region region-name \
```

- --node-type instance-type \
- --nodes-min 2 \
- --nodes-max 2 \
- --zones <AZ-1>,<AZ-2>

Using this

eksctl create cluster --name valaxy-cluster \

- --region ap-south-1 \
- --node-type t2.small \

AWS Dashboard -->> Cludformation -->> see valaxy cluster

RUN k8s basics command.

```
[root@developer2-system ~]# kubectl get nodes
                                               STATUS
                                                        ROLES
                                                                 AGE
                                                                       VERSION
NAME
                                               Ready
Ready
ip-192-168-17-254.ap-south-1.compute.internal
                                                                 17h
                                                                       v1.29.3-eks-ae9a62a
ip-192-168-59-94.ap-south-1.compute.internal
                                                                 16h
                                                                       v1.29.3-eks-ae9a62a
[root@developer2-system ~]#
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get all
                                CLUSTER-IP
                                             EXTERNAL-IP
                    TYPE
                                                           PORT(S)
                                                                     AGE
                    ClusterIP
                                                                     17h
service/kubernetes
                                                           443/TCP
NAME
default
                          17h
                 Active
kube-node-lease
                 Active
                          17h
                          17h
kube-public
                 Active
kube-system Active
[root@developer2-system ~]#
                          17h
[root@developer2-system ~]#
[root@developer2-system ~]# 🛮
```

kubectl get nodes

kubectl get all

kubectl get namespace

kubectl run webapp --image=httpd

kubectl get all

kubectl get po

kubectl version

```
[root@developer2-system ~]# kubectl run webapp --image=httpd
pod/webapp created
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get po
NAME
         READY
                 STATUS
                                      RESTARTS
                                                 AGE
webapp
         0/1
                 ContainerCreating
                                                 6s
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get po
         READY
                 STATUS
NAME
                           RESTARTS
                                       AGE
webapp
         1/1
                 Running
                                       9s
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get all
NAME
             READY
                     STATUS
                                RESTARTS
                                           AGE
pod/webapp
             1/1
                     Running
                                           19s
NAME
                     TYPE
                                  CLUSTER-IP
                                                             PORT(S)
                                                                        AGE
                                               EXTERNAL-IP
service/kubernetes
                     ClusterIP
                                  10.100.0.1
                                                              443/TCP
                                                                        17h
[root@developer2-system ~]#
[root@developer2-system ~]#
[root@developer2-system ~]#
```

Create Deployment using command line

kubectl create deployment demo-nginx --image=nginx --port=80 --replicas=2

kubectl get deployment

kubectl get pod

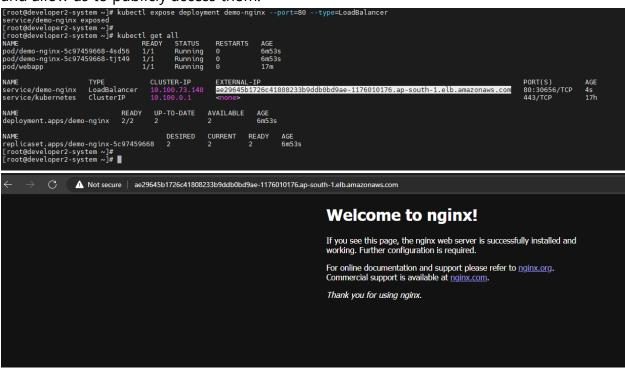
kubectl get all

kubectl expose deployment demo-nginx --port=80 --type=LoadBalancer

kubectl get all

```
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl create deployment demo-nginx --image=nginx --port=80 --replicas=2
deployment_apps/demo-nginx created
1/2
Gemo-ngthx 1/2
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get deploy
[root@developer2-system ~]# AVAILABLE AGE
                 READY UP-TO-DATE AVAILABLE
demo-nginx 2/2 2 2 17s
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get replicaset
NAME DESIRED CURRENT READY
                                DESIRED CURRENT
                                                                       AGE
demo-nginx-5c97459668 2 2
[root@developer2-system ~]#
[root@developer2-system ~]# kubectl get pod
                                                                       53s
                                         READY
                                                    STATUS
                                                                 RESTARTS
                                                                                 AGE
demo-nginx-5c97459668-4sd56
                                         1/1
                                                    Running
                                                                                 2m7s
demo-nginx-5c97459668-tjt49
                                         1/1
                                                    Running
                                                                                 2m7s
| 1/1 Runni
| [root@developer2-system ~]#
| [root@developer2-system ~]# kubectl get all
| NAME
                                         1/1
                                                                 0
                                                    Running
                                                                                 12m
                                                         STATUS
                                                                       RESTARTS
                                                                                      AGE
pod/demo-nginx-5c97459668-4sd56
                                                         Running
                                                                                      2m18s
pod/demo-nginx-5c97459668-tjt49
                                                         Running
                                                                                      2m18s
pod/webapp
                                                         Running
NAME
                             TYPE
                                             CLUSTER-IP
                                                              EXTERNAL-IP
                                                                                  PORT(S)
                                                                                               AGE
service/kubernetes
                             ClusterIP
                                                                                  443/TCP
                                             10.100.0.1
                                                                                               17h
                                                              <none>
                                                  UP-TO-DATE
                                       READY
                                                                    AVAILABLE
                                                                                     AGE
deployment.apps/demo-nginx
                                       2/2
                                                   2
                                                                                     2m18s
NAME
                                                       DESIRED
                                                                    CURRENT
                                                                                  READY
                                                                                             AGE
replicaset.apps/demo-nginx-5c97459668
[root@developer2-system ~]# _
                                                                                             2m18s
[root@developer2-system ~]# 🛮
```

Expose the deployment as service. This will create an ELB in front of those 2 containers and allow us to publicly access them.



Create First manifest file for pod and service

```
apiVersion: v1
kind: Service
metadata:
   name: demo-service
spec:
   ports:
   - name: nginx-port
    port: 80
     targetPort: 80

type: LoadBalancer
```

```
[root@developer2-system ~]# vim pod.yaml
[root@developer2-system ~]#
[root@developer2-system ~]# cat pod.yaml
apiVersion: v1
kind: Pod
metadata:
 name: demo-pod
  labels:
    app: demo
spec:
  containers:
    - name: demo-nginx
      image: nginx
     ports:
        - name: demo-ngnix
          containerPorts: 80
[root@developer2-system ~]#
[root@developer2-system ~]# vim service.yaml
[root@developer2-system ~]#
[root@developer2-system ~]#
[root@developer2-system ~]# cat service.yaml
apiVersion: v1
kind: Service
metadata:
 name: demo-service
spec:
  ports:
  - name: nginx-port
    port: 80
    targetPort: 80
  type: LoadBalancer
[root@developer2-system ~]# 📕
```

Create pod and service try to acces from browser it's not working for that we need to add label in pod yaml and selector in service yaml