Cluster Creation Time : 15 Min

Node Group Creation Time: 5 Min

<https://www.eksworkshop.com/beginner/110_irsa/>

##Install AWS CLI or Choose AWS EC2 Amazon Linux 2

Run aws configure and provide access and secret key.

##Kubectl

curl -LO "https://storage.googleapis.com/kubernetes-release/release/$(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt)/bin/linux/amd64/kubectl"

sudo mv ./kubectl /usr/local/bin/kubectl

chmod 700 /usr/local/bin/kubectl

kubectl version

##Download eksctl

curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp

sudo mv /tmp/eksctl /usr/local/bin

sudo chmod 700 /usr/local/bin/eksctl

eksctl version

https://documentation.sisense.com/latest/linux/prepeks.htm#gsc.tab=0

## aws configure

aws configure

AWS\_REGION=$(aws configure get region)

AWS\_REGION=us-east-1

eksctl create cluster \

--name "k8sb02-cluster" \

--version 1.18 \

--zones=us-east-1a,us-east-1b,us-east-1c \

--without-nodegroup

eksctl utils associate-iam-oidc-provider \

--region us-east-1 \

--cluster k8sb02-cluster \

--approve

aws eks describe-cluster --name engk8sb01-cluster --query cluster.identity.oidc.issuer --output text

#For Node Group In Public Subnet

eksctl create nodegroup --cluster=k8sb02-cluster \

--region=us-east-1 \

--name=k8sb02-cluster-ng-1 \

--node-type=t3.medium \

--nodes=2 \

--nodes-min=2 \

--nodes-max=4 \

--node-volume-size=20 \

--ssh-access \

--ssh-public-key=LaptopKey \

--managed \

--asg-access \

--external-dns-access \

--full-ecr-access \

--appmesh-access \

--alb-ingress-access

#For Node Group In Private

eksctl create nodegroup --cluster=k8sb02-cluster \

--region=us-east-1 \

--name=k8sb02-cluster-ng-2 \

--node-type=t3.medium \

--nodes=1 \

--nodes-min=2 \

--nodes-max=4 \

--node-volume-size=20 \

--ssh-access \

--ssh-public-key=LaptopKey \

--managed \

--asg-access \

--external-dns-access \

--full-ecr-access \

--appmesh-access \

--alb-ingress-access \

--node-private-networking

# List EKS Clusters

eksctl get clusters

# Capture Node Group name

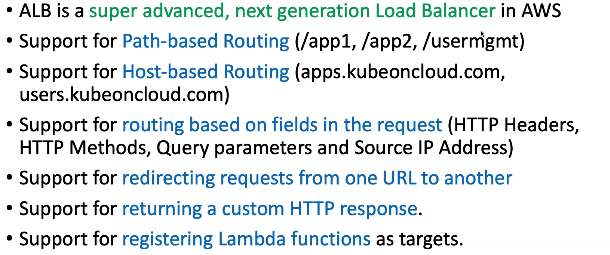
eksctl get nodegroup --cluster=<clusterName>

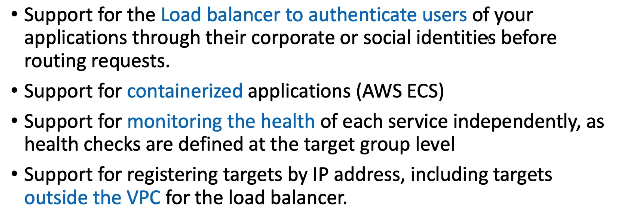
# Delete Node Group

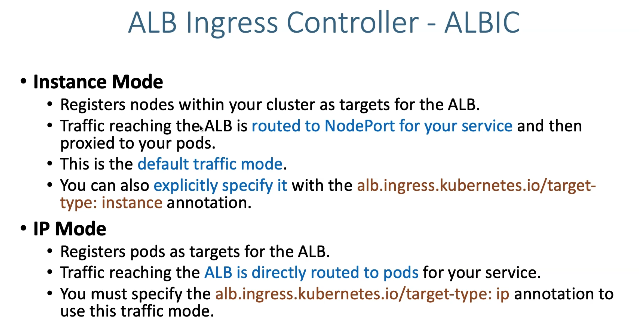
eksctl delete nodegroup --cluster=<clusterName> --name=<nodegroupName>

# Delete Cluster

eksctl delete cluster --name=<clusterName>







FARGATE dont support nodeport as it is serverless.

