* There is three way to use EKS in aws.

1. webUI
2. CLI
   1. Aws eks
   2. Eksctl
      1. This command provides some extra services.
3. API (terraform)

**Eksctl**

🡪Download it from google

🡪Copy in minikube folder (You can copy anywhere).

🡪Add to the path variable.

🡪Create IAm with admin power for using this.

🡪why to use eksctl?

🡪 aws eks command don’t have so much option.

🡪using this command, you cannot do lot of customization.

🡪suppose you need 2 t2.micro and 1 t2.small for worker node.This thing you can’t do using aws eks command.

🡪For doing this setup you have to tell how many nodes and how many nodegroup you need.

🡪node group is group of same nodes.

🡪One node group for 2 t2.micro, one group for 1 t2.small

🡪notepad “cluster.yml”

🡪eksctl create cluster -f cluster.yml

🡪They will automatically launch complete cluster for you.

🡪eksctl internally creating CloudFormation program.

🡪This program launch cluster.

🡪CloudFormation program is known as stack.

🡪In aws you want to automate anything you have to use cloud formation.

🡪This entire cluster launched in mumbai.

🡪Here eksctl try to launch 3 worker nodes in 3 different datacenter.

🡪For local connect we have to use kubectl.

🡪Here aws fully manage master node.

🡪So, you don’t have to worry about disaster.

🡪They will launch master in multiple AZ.

🡪aws eks update-kubeconfig --name akcluster

🡪This command will update the file and if there is no file then this command create config file for you.

🡪In your root folder, inside .kube file.

🡪Now you can use Kubectl from local system or any other system.

🡪aws get nodes

🡪kubectl describe nodes <id>

🡪kubectl create namespace akshit

🡪kubectl get ns

🡪This provides different and isolated environment to different teams.

🡪kubectl config set-context --current --namespace=akshit

🡪Now namespace changed to akshit

🡪Kubectl create deployment myweb --image=vimal13/apache-webserver-php

🡪Kubectl scale deployment myweb --replicas=3

🡪Now it will launch three pods.

🡪we use deployment so whenever any pod fail, it will create one more pod.

🡪If you use load balancer service and LoadBalancer type they know they have external load balancer ELB.

🡪kubectl expose deployment <myweb> --type=LoadBalancer --port=80.

🡪as soon as you run this command one ELB launch for you.

🡪ELB can do two things.

🡪Do the load balancing.

🡪Provide public IP.

🡪Here you don’t have to configure ELB.

🡪Here internally when you connect using public id then first it connects to ELB, ELB connect to the kubenetes (EKS) LoadBalancer and then connect to pods.

🡪kubectl describe service/myweb

🡪kubectl cp index.php <myweb-89ssdbnmdbz:/var/www/html/index.php>

🡪For using persistent storage, we have to use kubernetes PVC concept.

🡪Now a days we use dynamic PVC.

🡪storage class is already created by EKS.

🡪It by default takes storage from EBS.

🡪By default, created SC name is gp2 (Volume type).

🡪notepad pvc.yml

🡪kubectl create -f pvc.yml

🡪Here pvc created.

🡪Here as soon as pod request for PVC, PV created.

🡪Here SC is created by EKS and it is mentioned in EKS that volumeBindingMode=waitForFirstConsumer so, whenever pod request for PVC then only PV created.

🡪Consumer=pod

🡪kubectl get pv

🡪To check PV is not created.

🡪kubectl edit deployment <myweb>

🡪Edit command on the fly edit content of working pod.

🡪This will open yml file for you.

🡪Below spec, inside spec-part add this.

🡪volumes:

* name: web-vol1

persistentVolumeClaim:

claimName: akpvc

🡪In containers part, below image add this.

🡪volumeMounts:

* mountPath: /var/www/html

name: web-vol1

🡪save this file, you edit the file so, pods are recreated.

🡪this time PV created.

🡪kubectl get pv

🡪kubectl get sc

🡪Here, volume type = gp2, you can also create your own SC file.

🡪Here reclaimPolicy= DELETE

🡪So as soon as you delete PVC deleted, PV delete.

🡪and this deleted PV contact to SC and delete hard disk (EBS) also.

🡪While creating PVC we do not write storage class so by default it uses default SC.

🡪notepad “sc.yml”

🡪kubectl create –f sc.yml

🡪kubectl create –f pvc2.yml

🡪Here storage class mentioned,reclaimpolicy=retain.

🡪Here you don’t have use volumeBindingMode so as soon as you create pvc, pv created

#🡪kubectl edit sc gp2

# 🡪Copy the annotation part

#🡪kubectl edit sc aksc

# 🡪paste here inside metadata. You can also make this default, and #change whatever you want

#You can change this setting, but we want so I have comment it.

🡪eksctl delete cluster –f cluster.yml

-----------------------------------------------------------------------------------------------

🡪Aws launching node multiple different instance available.

1. On-demand

🡪By default, when you launch an instance in aws (ec2) it is on demand.

1. Spot instances.

🡪They are extra free resources (Free in their infrastructure.).

🡪aws provides you for sometimes.

🡪They are very cheaper.

🡪If they have spot instances then only they provide you.

🡪eksctl create cluster -f spot\_cluster.yml

🡺Why we need EKS?

🡪We can create configuration ourself also.

🡪But after deploy we have to manage it, provide resources, provide IO intensive HD, we don’t know how much we need in the future, external load balancer.

🡪Completely managed by aws-eks.

🡪EKS is tightly coupled with aws other services.

🡪aws take care of master node.

🡪 Some Constraint:

🡪In the eks you can integrate with only AWS service (Eg, If you use LoadBalancer type, you can only use ELB service, you cannot use external or personal loadbalancer).

🡪You cannot integrate other cloud or personal resources.

🡪You can not use your local HD.

🡪There is specific type in ELB, for this we have to use annotation.

🡪Using annotation, we do some changes.

🡪AWS completely manage the master node but in slave node they are asking us how many ram-CPU HD needs.

🡪Here we are not managing slave nodes. But we are telling them what to do.

🡪we want this part also aws manage from launching Kubernetes cluster to decide how much ram-CPU give (all security, connectivity).

🡪Here on the demand they provide more ram, CPU, storage, slave.

🡪This kind of setup is known as serverless architecture.

🡪Farget (aws service) provides serverless architecture for serverless world.

🡪It is inside ECS.

🡪They run time create slave. As soon as demand came they launch slave.

🡪eksctl get fargateprofile --cluster akcluster

🡪eksctl scale nodegroup --cluster akcluster --name ng2 --nodes=3 --nodes-max=5

🡪Here we are scalling the nodes.

🡪First, we set max node=2, so changed maximum node capacity to 5 so we can add 3 nodes.

🡪Here new cluster we have so we have to update configuration file.

🡪aws eks update-kubeconfig --name akcluster

🡪ssh –i mykey111222.pem –l ec2-user 192.168.34.25

🡪sudo su – root

🡪ps -aux | grep kubelet

🡪ifconfig –a

🡪Here you can see total 4 network card there.

🡪Kube schedular create two different containers in different node.

🡪But we need connectivity so they can contact to each other.

🡪In the Kubernetes world, clustering world we have third party programs, so we can connect two different vms.

🡪It is called interface.

🡪Here known as CNI (Container Network Interface).

🡪In Kubernetes world we use flannel plugins.

🡪In eks they internally build this setup.

🡪They use amazon own plugin.

🡪VPC contains multiple subnets.

🡪we have to put instance inside subnet.

🡪Without subnet you cannot launch instance.

🡪Whenever you use eksctl command, it automatically launches a subnet for you.

🡪Here for connecting two pod we have to give one more IP to both the pods, it is known as secondary IP.

🡪Here, if you see ps -aux | grep kubelet

🡪You can not launch more than 4 pods (For t2.micro).

🡪This is limitation of instances.

🡪We have lot of ram still we can not create more than 12 pods (t2.small).

🡪This concept is about planning.

🡪notepad fargate\_cluster.yml

🡪They ask access to 2 namespaces.

🡪eksctl create cluster -f fargate\_cluster .yml

🡪Fargate don’t work in Mumbai propely.

🡪So, launch in singapore.

🡪(other things are below mentoned.)

🡪Eksctl is just an automatation program, they create a code and give to cloudformation. Cloudformation do all other things.

🡪In aws One service like to communicate with other service they require permission - power.

🡪Here we require two things

🡪One service wants to communicate other; this permission is known as role.

🡪You are using EKS - aws so IAm created for connect master (EKS) from local windows using kubectl.

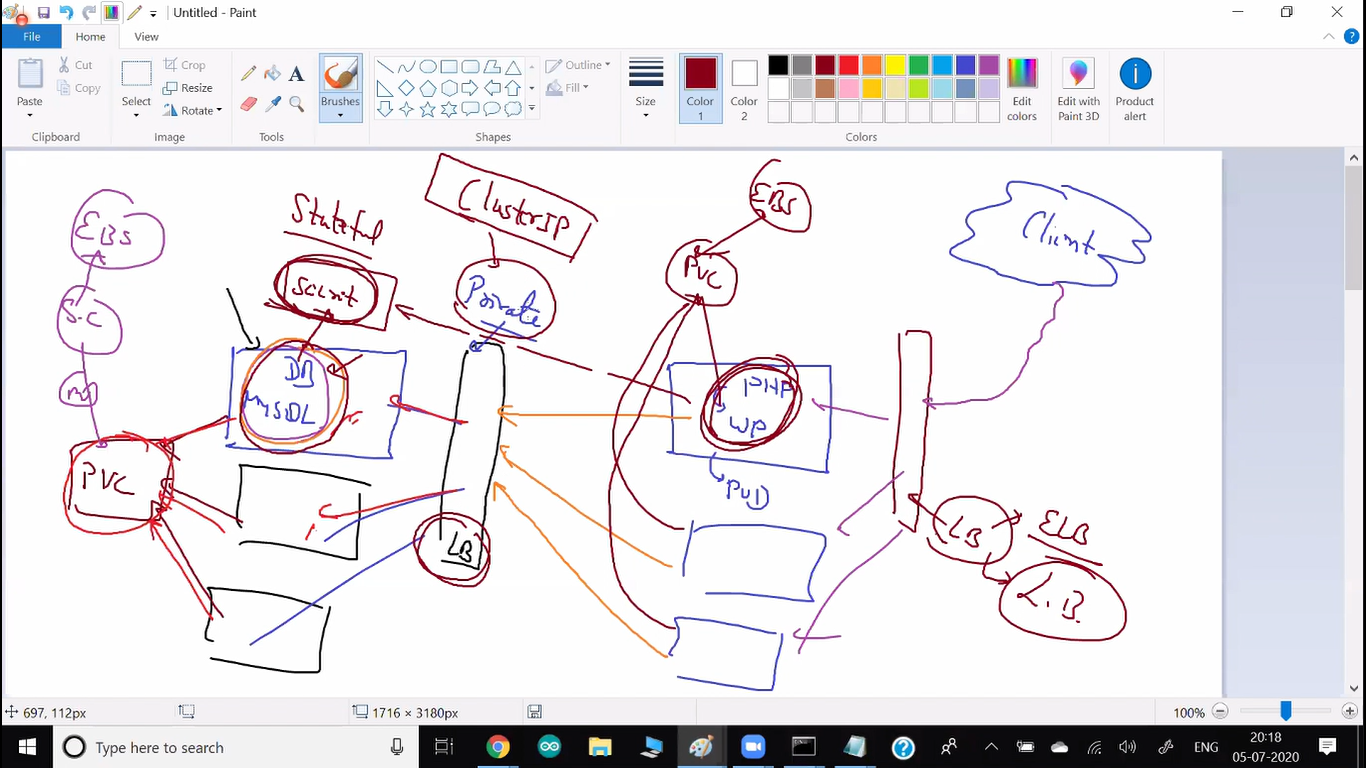
🡪For login to cluster require IAM, we give instruction to master.

🡪kubectl config view.

🡪Here you can see login details.

🡪Now master launch slave using EBS and for other service, master require role power.

* Wordpress-mysql configuration



🡪ReadWriteMany

🡪many pod can write in one single PVC.

🡪By default, they classic load balancer (It is free).

🡪create two files.

1. For wordpress (deploy to lb all)

🡪wordpress-deployment.yml

1. For mysql (all)

🡪mysql-deployment.yml

🡪Now we require to first launch database after that we have to launch wordpress.

🡪Without DB there is no meaning of wordpress.

🡪For this we use kustomization concept of Kubernetes.

🡪In this file first generate secret key for mysql pass and then first run mysql file then wp file.

🡪Kustomization concept used to run files in a sequence.

🡪kubectl create -k .

🡪Here we used load balancer (ELB).

🡪It gives DNS name.

🡪But whenever it restarts or something happened (It happen it rare cases only) next time this DNS name is changed.

🡪So, our client loose the connectivity.

🡪Instead of this you can write terraform file.

🡪In TF we can retrieve the DNS name

🡪In this set up we have one issue.

🡪We used EBS service but, one data centre EBS cannot be attached to other data centre EBS.

🡪For this we can use EFS concept.

🡪You have to launch EFS in same VPC and SG is also same.

🡪HELM is package manager for kubernetes.

🡪Here instead of package we use term chart.

🡪Install helm and tiller.

🡪helm init.

🡪They initialize helm for you.

🡪helm repo add stable <https://kubernetes-charts.storage.googleapis.com>

🡪helm repo list

🡪hlml search -l

🡪they show you all the packages.

🡪kubectl create ns akshit

🡪helm install --name my-release stable/jenkins

🡪It first requires tiller account and power.

🡪kubectl -n kube-system create se rviceaccount tiller

🡪kubectl create clusterrolebinding tiller --clusterrole cluster-admin --serviceaccount=kube-system:tiller

🡪helm init --service-account tiller --upgrade

🡪kubectl get pods -n kube-system

🡪 helm install --name my-release stable/jenkins

🡪refer doc for installing Prometheus.

🡺Fargate

🡪notepad fargate\_cluster.yml

🡪They ask access to 2 namespaces.

🡪eksctl create cluster -f fargate\_cluster .yml

🡪Fargate don’t work in Mumbai propely.

🡪So, launch in singapore.

🡪aws eks --region ap-southeast-1 update-kubeconfig --name far-akcluster

* aws eks help
* Aws eks list-clusters
* Aws eks create-cluster

🡪This command doesn’t have so much option.

🡪using these commands, you cannot do lot of customization.

* Eksctl version

🡪It behind the seen use aws login details.

🡪To check this use aws configure command.

* Eksctl get cluster
* eksctl create cluster -f cluster.yml
* aws eks update-kubeconfig --name akcluster

🡪This will automatically update config file for Kubectl.

🡺 kubectl create namespace akshit

🡺kubectl get ns

* Kubectl create deployment myweb --image=vimal13/apache-webserver-php
* Kubectl get pods –o wide

🡪This gives some extra details – where node is running.

* Kubectl scale deployment myweb --replicas=3
* kubectl expose deployment <myweb> --type=LoadBalancer --port=80
* kubectl describe service/myweb
* kubectl cp index.php <myweb-89ssdbnmdbz:/var/www/html/index.php>
* kubectl get sc
* kubectl edit sc aksc
* eksctl delete cluster –f cluster.yml

-----------------------------------------------------------------------------------------------

* eksctl get fargateprofile --cluster akcluster

🡪Launch fargate in Singapore, it not work propely in mumbai.

* eksctl scale nodegroup –cluster akcluster --name ng2 --nodes=3
* eksctl scale nodegroup –cluster akcluster --name ng2 --nodes=3 --nodes-max=5
* aws eks update-kubeconfig --name akcluster
* ssh –i mykey111222.pem –l ec2-user 192.168.34.25
  + sudo su – root
  + ps -aux | grep kubelet
  + ifconfig -a
* Kubectl create -k .