**Kubernetes interview questions :**

1. **What is Kubernetes:**

* It’s a container orchestration tool to maintain replicas, scaling etc.

1. Kubernetes Architecture:

**Master components :**

**Kube api :**

* auth - (username and password)
* authorization - (permission)
* code validation - api configurations
* pod creation request will go to kubeapi

**etcd db :**

* Kubernetes cluster database
* meta data (server details, masters node details, workers node details, all other components and all informations)
* pod status, deployment meta information - ip, name, volume, network

**Kubernetes Scheduler :**

      -    Watch

        -    It has an algorithm and it chooses which node is best for a container, it will check all the worker nodes

                  - It chooses the best node to deploy the application

**Kube controller :**

* Controller will not come in picture (until failure occurs)
* It maintain sub controllers - node controller / replica controller
* watch loop

**Worker Components:**

**Tv**

**Kubelet**

* Any provision request comes from master node(kube api) it will go to kubelet (it triggers docker to deploy and run the containers)
* status changes like POD running/ down will go to kube api and the records will be updated in etcd

**kube proxy**

* If a container moves to another worker node then kube proxy will contact another proxy and expose the container to the customer.
* It use layer 7 routing technology.
* Assume we have 3 worker nodes and if one worker node container wants to communicate to another pod container then kube proxy will help the communication.

**Docker (POD - pod of whales)**

Actually runs your application.

4. **Name space** - It is a project on top of kubernetes

5. **Pod**(EC2) - It is a compute resource for build containers - work load management

6.**service** -  without service we can’t communicate outside the pod or any internal communication.

**Types of services :**

Node port

Load balancer

cluster IP

Ingress

**Node port**

Pod to outer world communication or POD to POD communication. (It is a network component)

2. **Load balancer**

When it is a LB service , it will be mapped to EKS LB and it creates a VIP to communicate.

**3. Cluster ip**

Internal communication default service is called cluster IP.

**Ingress service:**

we are having 2 components in Ingress

* **Ingress controller** \_ it is the service which expose the pods to outer communication. He is the first service to handle 80,443 request.

eg: **nginx , traefik** etc

* **Ingress service** - It transfer to right POD or app based on the call!

* **HPA - Horizontal Pod Autoscaler**

* Horizontal scaling : Machine count or POD increase is horizontal scaling

**Liveness probe vs Readiness probe:**

* Liveness : It will restart the container if the expected health check is failed
* Readiness : It will stop the traffic to the containers until the container health is fine!

**What is CNI :**

* It’s a Container Network Interface which is used to communicate between worker nodes!

**Scenario Q/A:**

1. Architecture of the project?

* Answer them with you are supporting 2 products/ 3 / 4 /5 products.
* Each products will have Dev, QA, Pre-prod and production environments in different AWS accounts
* Each environments have single EKS clusters
* DEV - 6 /7 worker nodes
* Pre-prod/ Prod 10/ 11 worker nodes

1. Write a Docker file ?

FROM centos:7

MAINTAINER: Rahul gandhi

RUN yum update -y && yum install httpd -y

RUN echo “THis is test website” > /var/www/html

EXPOSE 80

CMD ‘/usr/sbin/httpd’

3. How to trigger git hub jobs to Jenkins

webhook - from git to jenkins job trigger

4. If a pod crashes what will you do ?

* Wil check the pod logs first using:

kubectl log pod/<podname> -n <namespace>

* If it is completely crashed try to redeploy it from jenkins pipeline
* Will check any recent application updates and will try to revert them
* Memory issue

5. Pod is up and running but when i hit the Load balancer i am getting a page not found error ?

* First we should check the POD locally using port-forward command
* kubectl port-forward pod/<podname> -n <namespace> 8080
* then in browser we should give the url as: localhost:8080 , if it is working locally
* then check the load balancer configuration and try to fix it

6. Explain me deployment flow from git to kubernetes

* we maintain our codes in Git lab and by using argo CD we automatically deploys the application to Kubernetes!

7. Replicaset vs replication controller

Replica set maintains at file level when it spins up the pod according to the value we mention in the file!

apiVersion: apps/v1

kind: Deployment

metadata:

 name: webdeployment

 namespace: devproject

spec:

 replicas: *2*

selector:

   matchLabels:

     apptype: web

     env: dev

 template:

   metadata:

     name: webtemp

     labels:

       apptype: web

       env: dev

Replication controller - It a process supervisor, mostly we use it for multipods across multiple nodes.

apiVersion: v1

kind: ReplicationController

metadata:

  name: nginx

spec:

  replicas: 3

  selector:

    app: nginx

  template:

    metadata:

      name: nginx

      labels:

        app: nginx

    spec:

      containers:

      - name: nginx

        image: nginx

        ports:

        - containerPort: 80

|  |  |
| --- | --- |
| **Replication Controller** | **Replica Set** |
| The Replication Controller is the original form of replication in Kubernetes | ReplicaSets are a higher-level API that gives the ability to easily run multiple instances of a given pod |
| The Replication Controller uses **equality-based selectors** to manage the pods. | ReplicaSets Controller uses **set-based selectors** to manage the pods. |
| The rolling-update command works with Replication Controllers | The rolling-update command won’t work with ReplicaSets. |
| Replica Controller is deprecated and replaced by ReplicaSets. | Deployments are recommended over ReplicaSets. |

8: How to terminate if pvc is stuck ?

Delete the PVC by using below command:

   kubectl delete pvc <pvcname> -n <name-space>

9, Node Affinity & antiAffinity

we can choose which node to deploy the pods are Node affinity