

KUBERNETES INTERVIEW QUESTIONS

1. What is the difference between Kubernetes and docker?

Docker is a container platform whereas Kubernetes is a container orchestration environment that offers capabilities like auto scaling, auto healing, clustering and enterprise level support like load balancing.

Or else

Docker is an opensource centralized platform designed to create, deploy and run applications. Docker uses container on the host OS to run applications. It allows applications to use same Linux kernel as a system on the host computer rather than creating a whole virtual OS. We can install docker on any OS but docker engine runs natively on Linux distributions. It is a tool that performs OS level virtualization also known as Containerization

Kubernetes is an open-source container management tool which automates container deployment, container scaling and load balancing. It schedules, runs and manages isolated containers which are running on virtual/ physical/ cloud machines. All top cloud providers support Kubernetes

2. What are the main components of Kubernetes architecture?

On a broad level, you can divide the k8s components in two parts:

A. Master/control plane

- I. API server
- II. ETCD
- III. Controller manager
- IV. Kube scheduler
- V. Cloud control manager (C-CM)

B. Data plane/ worker node

- I. Kube-let
- II. Kube-proxy
- III. Container runtime (docker, cri-o, containerD etc)

*You can explain each of the component by watching the video or by reading the official k8s document.

3. What is the main difference between the docker swarm and Kubernetes?

K8s is better suited for large organisations as it offers more scalability, networking capabilities like policies and huge third-party ecosystem.

FEATURES	KUBERNETES	DOCKER SWARM
Installation and cluster configuration	Complicated and time consuming	Fast and easy
Supports	K8s can work with almost all container types like rocket, docker, containerD	Work with docker only
GUI	GUI available	GUI not available
Data volumes	Only shared with containers in same POD	Can be shared with any other containers
Update and rollback	Process scheduling to maintain services while updating	Progressive updates and service health monitoring throughout the update
Autoscaling	Support vertical and horizontal autoscaling	Not support autoscaling
Logging and monitoring	Inbuilt tool present for monitoring	Used 3 rd party tools like splunk

4. What is the difference between docker container and Kubernetes pods?

A pod in k8s is a runtime specification of a container in docker. A pod provides more declarative way of defining using YAML and you can run, more than one container in a pod.

5. What is a namespace in Kubernetes?

In Kubernetes, namespace is a logical isolation of resources, network policies, RBAC and everything. For example, there are two projects using same k8s cluster. One project can use ns1 and other project can use ns2 without any overlap and authentication problems.

6. What is the role of kube-proxy?

Kube-proxy works by maintaining a set of network rules on each node in the cluster, which are updated dynamically as services are added or removed. When a client sends a request to a service, the request is intercepted by kube-proxy on the node where it was received. Kube-proxy then looks up the destination endpoint for the services and routes the request accordingly. Kube-proxy is an essential component of a Kubernetes cluster, as it ensures that services can communicate with each other.

7. What are the different types of services within Kubernetes?

- a. Cluster IP
- b. Node port
- c. Load balancing

8. What is the difference between node port and load balancer?

When a service is created with a node port type, the kube-proxy updates the IP tables with node-ip address and port that is chosen in the service configuration to access the pods. Whereas if you create a service as type load balancer, the cloud controller manager creates an external load balancer IP using the underlying cloud provider logic in the cloud controller manager. Users can access services using the external IP.

9. What is the role of kubelet?

Kubelet manages the containers that are scheduled to run on that node. It ensures that the containers are running and healthy and that the resources they need are available.

Kubelet communicates with the Kubernetes API server to get information about the containers that should be running on the node, and then starts and stops the containers as needed to maintain the desired state. It also monitors the containers to ensure that they are running correctly, and restarts them if necessary.

10.What are the day-to-day activities on Kubernetes?

- ✓ Devops engineers manages Kubernetes cluster on the organization.
- ✓ Ensures that the applications are running on different Kubernetes clusters should not have face any issues, if faces then fixes the issues and never let them to go down.
- ✓ Troubleshoot the bugs inside the cluster.
- ✓ Maintain different types of activities on cluster like installing necessary packages on the worker node, upgrade packages etc.
- ✓ Solve the issues by Jira ticket or any other ticketing.

*All above answers are not compulsory to tell, you can make your own, this is just copy from the video itself.

For more information subscribe the channel.

Abhishek Veerarmalla

Channel link: <https://www.youtube.com/@AbhishekVeeramalla/featured>

Video link: (Day-36)

<https://www.youtube.com/watch?v=0qy5JD7CbJ4&list=PLdpzxOOAlwvIKMhk8WhzN1pYoJ1YU8Csa&index=41>

DevOps play list:

<https://www.youtube.com/playlist?list=PLdpzxOOAlwvIKMhk8WhzN1pYoJ1YU8Csa>

Kubernetes play list:

https://www.youtube.com/playlist?list=PLdpzxOOAlwvJdsW6A0jCz_3VaANuFMLpc

Note Written by:

NAGARJUNA HOTA