

Top 30 Kubernetes Interview Questions and Answers

Top 30 Kubernetes Interview Questions and Answers

By [Simplilearn](#) Last updated on Jan 13, 2020 1573



Kubernetes comes from a Greek word meaning ‘captain,’ ‘helmsman,’ or ‘governor.’ The term is now also used in the DevOps and on-premises software development world to refer to a powerful bundle of solutions that equips operations engineers to scale and service server (and box) setups effortlessly. [Kubernetes](#) was created by Joe Beda, Craig McLuckie, and Brendan Burns, who were later joined by Google engineers before officially releasing it in 2014. Today, Kubernetes is maintained by Cloud

Native Computing Foundation (CNCF) and has evolved into a fast-growing and widely used ecosystem.

*Enhance your Kubernetes skills and gain credibility in the field with the [Certified Kubernetes Administrator Certification Training](#).
Enroll now!*

By now, you're probably wondering, what does Kubernetes do? Well, the answer to this question can't be fully explained within the scope of this article. After all, there are entire Kubernetes courses that are designed to answer this question, including how to use it.

However, what we'll cover here are some frequently asked Kubernetes interview questions and answers. These questions and answers will help you prepare for any interview or certification exam that you may need to take once you've completed the Kubernetes training. So, without further ado, let's jump right in.

Also Read: [Best DevOps Tools to Learn and Master](#) for more information on the best DevOps tools like Docker, Kubernetes, Ansible, Git, and Puppet.

Kubernetes Interview Questions and Answers

1. What is Kubernetes?

[Kubernetes](#) is an open-source container orchestration tool or system that is used to automate tasks such as the management, monitoring,

scaling, and deployment of containerized applications. It is used to easily manage several containers (since it can handle grouping of containers), which provides for logical units that can be discovered and managed.

2. What are K8s?

K8s is another term for Kubernetes.

3. What is orchestration when it comes to software and DevOps?

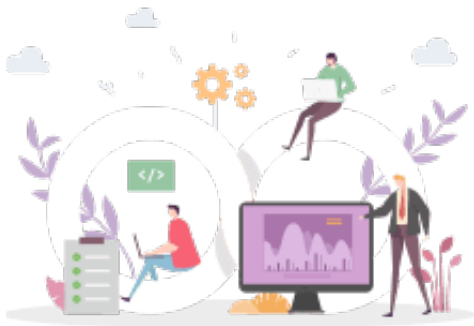
Orchestration refers to the integration of multiple services that allows them to automate processes or synchronize information in a timely fashion. Say, for example, you have six or seven microservices for an application to run. If you place them in separate containers, this would inevitably create obstacles for communication. Orchestration would help in such a situation by enabling all services in individual containers to work seamlessly to accomplish a single goal.

Offer Expires In

- 00 : HRS
- 35 : MIN
- 06SEC

DevOps Career Guide

A Guide to Becoming A DevOps Engineer[DOWNLOAD GUIDE](#)



4. How are Kubernetes and Docker related?

Docker is an open-source platform used to handle software development. Its main benefit is that it packages the settings and dependencies that the software/application needs to run into a container, which allows for portability and several other advantages. Kubernetes allows for the manual linking and orchestration of several containers, running on multiple hosts that have been created using Docker.

5. What are the main differences between the Docker Swarm and Kubernetes?

Docker Swarm is Docker's native, open-source container orchestration platform that is used to cluster and schedule Docker containers. Swarm differs from Kubernetes in the following ways:

- Docker Swarm is more convenient to set up but doesn't have a robust cluster, while Kubernetes is more complicated to set up but

the benefit of having the assurance of a robust cluster

- Docker Swarm can't do auto-scaling (as can Kubernetes); however, Docker scaling is five times faster than Kubernetes
- Docker Swarm doesn't have a GUI; Kubernetes has a GUI in the form of a dashboard
- Docker Swarm does automatic load balancing of traffic between containers in a cluster, while Kubernetes requires manual intervention for load balancing such traffic
- Docker requires third-party tools like ELK stack for logging and monitoring, [while Kubernetes](#) has integrated tools for the same
- Docker Swarm can share storage volumes with any container easily, while Kubernetes can only share storage volumes with containers in the same pod
- Docker can deploy rolling updates but can't deploy automatic rollbacks; Kubernetes can deploy rolling updates as well as automatic rollbacks

6. What are the main components of Kubernetes architecture?

There are two primary components: the master node and the worker node. Each of these components has individual components in them.

7. What is a node in Kubernetes?

A node is the smallest fundamental unit of computing hardware. It represents a single machine in a cluster, which could be a physical machine in a data center or a virtual machine from a cloud provider. Each machine can substitute any other machine in a Kubernetes cluster. The master in Kubernetes controls the nodes that have containers.

8. What does the node status contain?

The main components of a node status are Address, Condition, Capacity, and Info.

9. What process runs on Kubernetes Master Node?

The Kube-api server process runs on the master node and serves to scale the deployment of more instances.

10. What is a pod in Kubernetes?

Pods are high-level structures that wrap one or more containers. This is because containers are not run directly in Kubernetes. Containers in the same pod share a local network and the same resources, allowing them to easily communicate with other containers in the same pod as if they were on the same machine while at the same time maintaining a degree of isolation.

11. What is the job of the kube-scheduler?

The kube-scheduler assigns nodes to newly created pods.

12. What is a cluster of containers

in Kubernetes?

A cluster of containers is a set of machine elements that are nodes. Clusters initiate specific routes so that the containers running on the nodes can communicate with each other. In Kubernetes, the container engine (not the server of the Kubernetes API) provides hosting for the API server.

13. What is the Google Container Engine?

The Google Container Engine is an open-source management platform tailor-made for Docker containers and clusters to provide support for the clusters that run in Google public cloud services.

14. What are Daemon sets?

A Daemon set is a set of pods that runs only once on a host. They are used for host layer attributes like a network or for monitoring a network, which you may not need to run on a host more than once.

15. What is ‘Heapster’ in Kubernetes?

A Heapster is a performance monitoring and metrics collection system for data collected by the Kubelet. This aggregator is natively supported and runs like any other pod within a Kubernetes cluster, which allows it to discover and query usage data from all nodes within the cluster.

16. What is a Namespace in Kubernetes?

Namespaces are used for dividing cluster resources between multiple users. They are meant for environments where there are many users spread across projects or teams and provide a scope of resources.

17. Name the initial namespaces from which Kubernetes starts?

- Default
- Kube – system
- Kube – public

18. What is the Kubernetes controller manager?

The controller manager is a daemon that is used for embedding core control loops, garbage collection, and Namespace creation. It enables the running of multiple processes on the master node even though they are compiled to run as a single process.

DevOps Engineer Master's Program

Bridge between software developers and operations [Explore Course](#)



19. What are the types of controller managers?

The primary controller managers that can run on the master node are the endpoints controller, service accounts controller, namespace controller, node controller, token controller, and replication controller.

20. What is etcd?

Kubernetes uses etcd as a distributed key-value store for all of its data, including metadata and configuration data and allows nodes in Kubernetes clusters to read and write data. Although etcd was purposely built for CoreOS, it also works on a variety of operating systems (e.g., Linux, BSD, and OS X) because it is open-source. Etcd represents the state of a cluster at a specific moment in time and is a canonical hub for state management and cluster coordination of a Kubernetes cluster.

21. What are the different services within Kubernetes?

Different types of Kubernetes services include:

- Cluster IP service
- Node Port service
- External Name Creation service and
- Load Balancer service

22. What is ClusterIP?

The ClusterIP is the default Kubernetes service that provides a service inside a cluster (with no external access) that other apps inside your cluster can access.

23. What is NodePort?

The NodePort service is the most fundamental way to get external traffic directly to your service. It opens a specific port on all Nodes and forwards any traffic sent to this port to the service.

24. What is the LoadBalancer in Kubernetes?

The LoadBalancer service is used to expose services to the internet. A Network load balancer, for example, creates a single IP address that forwards all traffic to your service.

25. What is a headless service?

A headless service is used to interface with service discovery

mechanisms without being tied to a ClusterIP, therefore allowing you to directly reach pods without having to access them through a proxy. It is useful when neither load balancing nor a single Service IP is required.

26. What is a kubelet?

The kubelet is a service agent that controls and maintains a set of pods by watching for pod specs through the Kubernetes API server. It preserves the pod lifecycle by ensuring that a given set of containers are all running as they should. The kubelet runs on each node and enables the communication between the master and slave nodes.

27. What is Kubectl?

Kubectl is a CLI (command-line interface) that is used to run commands against Kubernetes clusters. As such, it controls the Kubernetes cluster manager through different create and manage commands on the Kubernetes component

28. Give examples of recommended security measures for Kubernetes.

Examples of standard Kubernetes security measures include defining resource quotas, support for auditing, restriction of etcd access, regular security updates to the environment, network segmentation, definition of strict resource policies, continuous scanning for security vulnerabilities, and using images from authorized repositories.

29. What is Kube-proxy?

Kube-proxy is an implementation of a load balancer and network proxy used to support service abstraction with other networking operation. Kube-proxy is responsible for directing traffic to the right container based on IP and the port number of incoming requests.

30. How can you get a static IP for a Kubernetes load balancer?

A static IP for the Kubernetes load balancer can be achieved by changing DNS records since the Kubernetes Master can assign a new static IP address.

Check out the following video to get an in-depth understanding of the Kubernetes interview questions and answers -

Jumpstart Your Career with Kubernetes Training

Having a good [understanding of DevOps](#) and on-premises [software development](#) can be quite useful in helping you gain a holistic view of the subject matter. Ultimately, taking the [Certified Kubernetes Administrator Course](#) and taking your time to study and understand what you've learned while preferably putting it into practice is the best way to prepare for an interview. The questions here are the icing on the cake because they will allow you to get a feel of the type of questions that you can be asked. The more familiar you are with these types of questions, the better able you will be to show off your skills.

