

## **Kubernetes Interview Questions**

### **1.What is kubernetes?**

This is one of the most basic Kubernetes interview questions yet one of the most important ones! 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 is POD in kubernetes? How it differs from Container?

3. How to make node un-schedulable in kubernetes?

4.What is a JOB in kubernetes?

5.How does service discovery happen in kubernetes?

6.How do you interact with kubernetes cluster from outside? which server is responsible for it?

7.What is storage classes in kubernetes?

8.What is the difference between stateful sets and daemon sets in kubernetes?

9.Can we have external Endpoint for service in kubernetes?

10. 3 containers are running; one container is consuming all the cpu memory (99%) will it be caused any issue to remaining containers? If yes, what is the solution?

11.We are running on one container and that got failed. What is the possible ways to rectify that and what are the steps to take further?

12.What is the difference between kubctl & kops?

13.Why you are using kubectl can you explain why we are using?

**For Online / Classroom Training Call : +91- 9611140077**

**[www.pragathitech.com](http://www.pragathitech.com)**

14. Why are we using kops?
15. How many projects you used kubernetes?
16. number of docker containers deployed to different vm's how will you manage there is no kubernetes installed?
17. If installed kubernetes how you will deploy this container into kubernetes cluster?
18. Can you tell me the command for creating kubernetes cluster in vm?
19. How many nodes we required to create kubernetes cluster?
20. We have nearly 15 nodes in my organization all are decentralized so which node I need to create as a master? Is there any possibility to make all the machines as masters?
21. Our applications are decentralized I don't want distributed environment if anything happens to the master all will collapse, can we create multiple masters?
22. What is the difference between kubect1 and minikube?
23. If any container down in my cluster how you will rectify?
24. How to attach a volume in cluster at some time the container will be deleted then rs will re-create new container then how to attach that container automatically and how to restore the volume automatically to re-created container?
25. What you will do if one master got corrupted, can we create multiple masters?
26. What you will do in case any pod deleted?
27. What is namespaces in kubernetes can you tell me some?
28. Can you tell me some commands using in kubernetes?
29. Tell me the command to create cluster?

**For Online / Classroom Training Call : +91- 9611140077**

**[www.pragathitech.com](http://www.pragathitech.com)**

30. Difference between rc and rs?

### **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.

### **How are Kubernetes and Docker related?**

This is one of the most frequently asked Kubernetes interview questions, where the interviewer might as well ask you to share your experience working with any of them. 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.

### **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

**For Online / Classroom Training Call : +91- 9611140077**

**[www.pragathitech.com](http://www.pragathitech.com)**

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

### **What is the difference between deploying applications on hosts and containers?**

Deploying Applications consist of an architecture that has an operating system. The operating system will have a kernel that holds various libraries installed on the operating system needed for an application.

Whereas container host refers to the system that runs the containerized processes. This kind is isolated from the other applications; therefore, the applications must have the necessary libraries. The binaries are separated from the rest of the system and cannot infringe any other application.

### **What are the features of Kubernetes?**

Kubernetes places control for the user where the server will host the container. It will control how to launch. So, Kubernetes automates various manual processes.

Kubernetes manages various clusters at the same time.

It provides various additional services like management of containers, security, networking, and storage.

Kubernetes self-monitors the health of nodes and containers.

With Kubernetes, users can scale resources not only vertically but also horizontally that too easily and quickly.

### **What are the main components of Kubernetes architecture?**

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

**Explain the working of the master node in Kubernetes?**

The master node dignifies the node that controls and manages the set of worker nodes. This kind resembles a cluster in Kubernetes. The nodes are responsible for the cluster management and the API used to configure and manage the resources within the collection. The master nodes of Kubernetes can run with Kubernetes itself, the asset of dedicated pods.

**What is the role of Kube-apiserver?**

This kind validates and provides configuration data for the API objects. It includes pods, services, replication controllers. Also, it provides REST operations and also the frontend of the cluster. This frontend cluster state is shared through which all other component interacts.

**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.

**What does the node status contain?**

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

**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.

**What is a pod in Kubernetes?**

In this Kubernetes interview question, try giving a thorough answer instead of a one-liner. 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.

**What is the job of the kube-scheduler?**

The kube-scheduler assigns nodes to newly created pods.

**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.

**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.

**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.

**What is 'Heapster' in Kubernetes?**

In this Kubernetes interview question, the interviewer would expect a thorough explanation. You can explain what it is and also it has been useful to you (if you have used it in your work so far!). 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.

**What is Minikube?**

With the help of Minikube, users can run Kubernetes locally. This process lets the user run a single-node Kubernetes cluster on your personal computer, including Windows, macOS, and Linux PCs. With this, users can try out Kubernetes also for daily development work.

**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.

**Name the initial namespaces from which Kubernetes starts?****Default**

Kube – system

Kube – public

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)



### **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.

### **What is Kubernetes?**

This is one of the most basic Kubernetes interview questions yet one of the most important ones! 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.

### **What are K8s?**

K8s is another term for Kubernetes.

### **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.

### **How are Kubernetes and Docker related?**

This is one of the most frequently asked Kubernetes interview questions, where the interviewer might as well ask you to share your experience working with any of them. 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.

### **What is the difference between deploying applications on hosts and containers?**

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

Deploying Applications consist of an architecture that has an operating system. The operating system will have a kernel that holds various libraries installed on the operating system needed for an application.

Whereas container host refers to the system that runs the containerized processes. This kind is isolated from the other applications; therefore, the applications must have the necessary libraries. The binaries are separated from the rest of the system and cannot infringe any other application.

### **What are the features of Kubernetes?**

Kubernetes places control for the user where the server will host the container. It will control how to launch. So, Kubernetes automates various manual processes.

Kubernetes manages various clusters at the same time.

It provides various additional services like management of containers, security, networking, and storage.

Kubernetes self-monitors the health of nodes and containers.

With Kubernetes, users can scale resources not only vertically but also horizontally that too easily and quickly.

### **What are the main components of Kubernetes architecture?**

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

### **Explain the working of the master node in Kubernetes?**

The master node dignifies the node that controls and manages the set of worker nodes. This kind resembles a cluster in Kubernetes. The nodes are responsible for the cluster management and the API used to configure and manage the resources within the collection. The master nodes of Kubernetes can run with Kubernetes itself, the asset of dedicated pods.

### **What is the role of Kube-apiserver?**

This kind validates and provides configuration data for the API objects. It includes pods, services, replication controllers. Also, it provides REST operations and also the frontend of the cluster. This frontend cluster state is shared through which all other component interacts.

### **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.

### **What does the node status contain?**

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

### **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.

### **What is a pod in Kubernetes?**

In this Kubernetes interview question, try giving a thorough answer instead of a one-liner. 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.

### **What is the job of the kube-scheduler?**

The kube-scheduler assigns nodes to newly created pods.

### **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.

### **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.

### **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.

### **What is 'Heapster' in Kubernetes?**

In this Kubernetes interview question, the interviewer would expect a thorough explanation. You can explain what it is and also it has been useful to you (if you have used it in your work so far!). 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.

### **What is Minikube?**

With the help of Minikube, users can run Kubernetes locally. This process lets the user run a single-node Kubernetes cluster on your personal computer, including Windows, macOS, and Linux PCs. With this, users can try out Kubernetes also for daily development work.

### **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.

### **Name the initial namespaces from which Kubernetes starts?**

#### **Default**

Kube – system

Kube – public

### **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.

### **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.

### **What is etcd?**

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

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, BSB, 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.

### **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

### **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.

### **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.

### **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.

### **What is the Ingress network, and how does it work?**

An ingress is an object that allows users to access your Kubernetes services from outside the Kubernetes cluster. Users can configure the access by creating rules that define which inbound connections reach which services.

### **What is the difference between deploying applications on hosts and containers?**

Deploying Applications consist of an architecture that has an operating system. The operating system will have a kernel that holds various libraries installed on the operating system needed for an application.

Whereas container host refers to the system that runs the containerized processes. This kind is isolated from the other applications; therefore, the applications must have the necessary libraries. The binaries are separated from the rest of the system and cannot infringe any other application.

### **What does the node status contain?**

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

### **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.

### **What is a pod in Kubernetes?**

In this Kubernetes interview question, try giving a thorough answer instead of a one-liner. 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.

### **What is the job of the kube-scheduler?**

The kube-scheduler assigns nodes to newly created pods.

### **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.

### **What are Daemon sets?**

**For Online / Classroom Training Call : +91- 9611140077**

**[www.pragathitech.com](http://www.pragathitech.com)**

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.

### **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.

### **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

Free Course: Introduction to DevOps Tools

Master the Fundamentals of DevOps Tools ENROLL NOW Free Course:

Introduction to DevOps Tools

### **What do you understand by Cloud controller manager?**

You must have heard about Public, Private and hybrid clouds. With the help of cloud infrastructure technologies, you can run Kubernetes on them. In the context of Cloud Controller Manager, it is the control panel component that embeds the cloud-specific control logic. This process lets you link the cluster into the cloud provider's API and separates the elements that interact with the cloud platform from components that only interact with your cluster.

This also enables the cloud providers to release the features at a different pace compared to the main Kubernetes project. It is structured using a plugin mechanism and allows various cloud providers to integrate their platforms with Kubernetes.

### **What is Container resource monitoring?**

This refers to the activity that collects the metrics and tracks the health of containerized applications and microservices environments. It helps to

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

improve health and performance and also makes sure that they operate smoothly.

### **What is the difference between a replica set and a replication controller?**

A replication controller is referred to as RC in short. It is a wrapper on a pod. This provides additional functionality to the pods, which offers replicas.

It monitors the pods and automatically restarts them if they fail. If the node fails, this controller will respawn all the pods of that node on another node. If the pods die, they won't be spawned again unless wrapped around a replica set.

Replica Set, on the other hand, is referred to as rs in short. It is told as the next-generation replication controller. This kind of support has some selector types and supports the equality-based and the set-based selectors.

It allows filtering by label values and keys. To match the object, they have to satisfy all the specified label constraints.

### **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.

### **What are federated clusters?**

The aggregation of multiple clusters that treat them as a single logical cluster refers to cluster federation. In this, multiple clusters may be managed as a single cluster. They stay with the assistance of federated groups. Also, users can create various clusters within the data center or cloud and use the federation to control or manage them in one place.

You can perform cluster federation by doing the following:

Cross cluster that provides the ability to have DNS and Load Balancer with backend from the participating clusters.



Users can sync resources across different clusters in order to deploy the same deployment set across the various clusters.

With this, users can easily set up the rules for routing traffic without creating a bunch of load balancers or exposing each service to the nodes.

### **What is 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.

### **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

### **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.

### **What is Kube-proxy?**

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

### **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.

### **How to do maintenance activity on the K8 node?**

Whenever there are security patches available the Kubernetes administrator has to perform the maintenance task to apply the security patch to the running container in order to prevent it from vulnerability,

**For Online / Classroom Training Call : +91- 9611140077**

**[www.pragathitech.com](http://www.pragathitech.com)**

which is often an unavoidable part of the administration. The following two commands are useful to safely drain the K8s node.

```
kubectl cordon
```

```
kubectl drain --ignore-daemon set
```

The first command moves the node to maintenance mode or makes the node unavailable, followed by `kubectl drain` which will finally discard the pod from the node. After the drain command is a success you can perform maintenance.

Note: If you wish to perform maintenance on a single pod following two commands can be issued in order:

```
kubectl get nodes: to list all the nodes
```

```
kubectl drain <node name>: drain a particular node
```

### **How do we control the resource usage of POD?**

With the use of limit and request resource usage of a POD can be controlled.

**Request:** The number of resources being requested for a container. If a container exceeds its request for resources, it can be throttled back down to its request.

**Limit:** An upper cap on the resources a single container can use. If it tries to exceed this predefined limit it can be terminated if K8's decides that another container needs these resources. If you are sensitive towards pod restarts, it makes sense to have the sum of all container resource limits equal to or less than the total resource capacity for your cluster.

```
apiVersion: v1
```

```
kind: Pod
```

```
metadata:
```

```
  name: demo
```

```
spec:
```

```
  containers:
```

```
    - name: example1
```

```
      image: example/example1
```

```
      resources:
```

```
        requests:
```

```
          memory: "1Mi"
```

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

```
cpu: "_m"  
limits:  
  memory: "_Mi"  
  cpu: "_m"
```

**What are the various K8's services running on nodes and describe the role of each service?**

Mainly K8 cluster consists of two types of nodes, executor and master.

Executor node: (This runs on master node)

Kube-proxy: This service is responsible for the communication of pods within the cluster and to the outside network, which runs on every node. This service is responsible to maintain network protocols when your pod establishes a network communication.

kubelet: Each node has a running kubelet service that updates the running node accordingly with the configuration(YAML or JSON) file.

NOTE: kubelet service is only for containers created by Kubernetes.

Master services:

Kube-apiserver: Master API service which acts as an entry point to K8 cluster.

Kube-scheduler: Schedule PODs according to available resources on executor nodes.

Kube-controller-manager: is a control loop that watches the shared state of the cluster through the apiserver and makes changes attempting to move the current state towards the desired stable state

**What is PDB (Pod Disruption Budget)?**

A Kubernetes administrator can create a deployment of a kind: PodDisruptionBudget for high availability of the application, it makes sure that the minimum number of running pods are respected as mentioned by the attribute minAvailable spec file. This is useful while performing a drain where the drain will halt until the PDB is respected to ensure the High Availability(HA) of the application. The following spec file also shows minAvailable as 2 which implies the minimum number of an available pod (even after the election).

Example: YAML Config using minAvailable =>

```
apiVersion: policy/v1beta1  
kind: PodDisruptionBudget
```

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

```
metadata:  
  name: zk-pdb  
spec:  
  minAvailable: 2  
  selector:  
    matchLabels:  
      app: zookeeper
```

### **What's the init container and when it can be used?**

init containers will set a stage for you before running the actual POD.

Wait for some time before starting the app Container with a command like sleep 60.

Clone a git repository into a volume.

### **What is the role of Load Balance in Kubernetes?**

Load balancing is a way to distribute the incoming traffic into multiple backend servers, which is useful to ensure the application available to the users.

If the incoming traffic lands to a single IP address on the load balancer which is a way to expose your service to outside the internet which routes the incoming traffic to a particular pod (via service) using an algorithm known as round-robin. Even if any pod goes down load balancers are notified so that the traffic is not routed to that particular unavailable node. Thus load balancers in Kubernetes are responsible for distributing a set of tasks (incoming traffic) to the pods

### **What are the various things that can be done to increase Kubernetes security?**

By default, POD can communicate with any other POD, we can set up network policies to limit this communication between the PODs.

RBAC (Role-based access control) to narrow down the permissions.

Use namespaces to establish security boundaries.

Set the admission control policies to avoid running the privileged containers.

Turn on audit logging.

### **How to monitor the Kubernetes cluster?**

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

Prometheus is used for Kubernetes monitoring. The Prometheus ecosystem consists of multiple components.

Mainly Prometheus server which scrapes and stores time-series data.  
Client libraries for instrumenting application code.  
Push gateway for supporting short-lived jobs.  
Special-purpose exporters for services like StatsD, HAProxy, Graphite, etc.  
An alert manager to handle alerts on various support tools.

### **How to get the central logs from POD?**

This architecture depends upon the application and many other factors. Following are the common logging patterns

Node level logging agent.  
Streaming sidecar container.  
Sidecar container with the logging agent.  
Export logs directly from the application.  
In the setup, journalbeat and filebeat are running as daemonset. Logs collected by these are dumped to the kafka topic which is eventually dumped to the ELK stack.

The same can be achieved using EFK stack and fluentd-bit.

### **Why use namespaces? What is the problem with using the default namespace?**

While using the default namespace alone, it becomes hard over time to get an overview of all the applications you can manage in your cluster. Namespaces make it easier to organize the applications into groups that make sense, like a namespace of all the monitoring applications and a namespace for all the security applications, etc.

Namespaces can also be useful for managing Blue/Green environments where each namespace can include a different version of an app and also share resources that are in other namespaces (namespaces like logging, monitoring, etc.).

Another use case for namespaces is one cluster with multiple teams. When multiple teams use the same cluster, they might end up stepping on each other's toes. For example, if they end up creating an app with the same name it means one of the teams overrides the app of the other

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)

team because there can't be two apps in Kubernetes with the same name (in the same namespace).

### **What is an Operator?**

"Operators are software extensions to K8s which make use of custom resources to manage applications and their components. Operators follow Kubernetes principles, notably the control loop."

### **Why do we need Operators?**

The process of managing applications in Kubernetes isn't as straightforward as managing stateless applications, where reaching the desired status and upgrades are both handled the same way for every replica. In stateful applications, upgrading each replica might require different handling due to the stateful nature of the app, each replica might be in a different status. As a result, we often need a human operator to manage stateful applications. Kubernetes Operator is supposed to assist with this.

This will also help with automating a standard process on multiple Kubernetes clusters

### **What is GKE?**

GKE is Google Kubernetes Engine that is used for managing and orchestrating systems for Docker containers. With the help of Google Public Cloud, we can also orchestrate the container cluster.

### **What is Ingress Default Backend?**

It specifies what to do with an incoming request to the Kubernetes cluster that isn't mapped to any backend i.e what to do when no rules being defined for the incoming HTTP request. If the default backend service is not defined, it's recommended to define it so that users still see some kind of message instead of an unclear error.

### **How to run Kubernetes locally?**

Kubernetes can be set up locally using the Minikube tool. It runs a single-node bunch in a VM on the computer. Therefore, it offers the perfect way for users who have just ongoing learning Kubernetes.

### **What is Kubernetes Load Balancing?**

**For Online / Classroom Training Call : +91- 9611140077**

[www.pragathitech.com](http://www.pragathitech.com)



Load Balancing is one of the most common and standard ways of exposing the services. There are two types of load balancing in K8s and they are:

Internal load balancer – This type of balancer automatically balances loads and allocates the pods with the required incoming load.

External Load Balancer – This type of balancer directs the traffic from the external loads to backend pods

[www.pragathitech.com](http://www.pragathitech.com)