**Basic Concepts**

1. **What is Kubernetes?**
   * **Answer:** Kubernetes is an open-source platform that automates deploying, scaling, and managing containerized applications. It provides tools for container orchestration, self-healing, load balancing, and more.
2. **What are the main components of Kubernetes?**
   * **Answer:** The key components of Kubernetes are:
     + **Control Plane:** API server, Scheduler, Controller Manager, etcd.
     + **Nodes:** Worker nodes that run Pods, Kubelet, Kube Proxy.
3. **What is a container?**
   * **Answer:** A container is a lightweight, standalone, executable package of software that includes everything needed to run an application (code, runtime, libraries).
4. **What is a Pod in Kubernetes?**
   * **Answer:** A Pod is the smallest deployable unit in Kubernetes, which can contain one or more containers. All containers in a pod share the same network and storage resources.
5. **What is a ReplicaSet in Kubernetes?**
   * **Answer:** A ReplicaSet ensures that a specified number of replicas of a pod are running at any given time. It maintains the desired state of the system and automatically replaces any failed pods.
6. **What is a Deployment in Kubernetes?**
   * **Answer:** A Deployment manages a set of replica pods. It provides features like rolling updates, rollback, and scaling to ensure application availability and stability.
7. **What is a Service in Kubernetes?**
   * **Answer:** A Service is an abstraction that defines a logical set of Pods and a policy to access them. Services provide stable endpoints for pod communication and load balancing.
8. **What are Namespaces in Kubernetes?**
   * **Answer:** Namespaces are virtual clusters within a Kubernetes cluster. They help in organizing and isolating resources for different users or teams.
9. **What is the Kubernetes Scheduler?**
   * **Answer:** The Kubernetes Scheduler is responsible for assigning pods to nodes based on resource availability, policies, and constraints.
10. **What is etcd in Kubernetes?**

* **Answer:** etcd is a distributed key-value store used by Kubernetes to store configuration data and the cluster's state.

**Pods and Containers**

1. **What is the difference between a Pod and a Container?**

* **Answer:** A Pod is a group of one or more containers that share the same network and storage. Containers in a pod are tightly coupled and always deployed together.

1. **What are Init Containers?**

* **Answer:** Init containers are special containers that run before the main application containers in a pod. They are used for initialization tasks such as setting up configurations or dependencies.

1. **How does Kubernetes handle Pod scheduling?**

* **Answer:** Kubernetes uses the scheduler to assign pods to nodes based on resource requirements, policies, and constraints.

1. **What is the use of a sidecar container?**

* **Answer:** A sidecar container is a helper container running alongside the main container in a pod. It often performs auxiliary tasks such as logging, monitoring, or proxying.

1. **How can you check if a pod is running?**

* **Answer:** Use the kubectl get pods command to check the status of pods. It will show the status as Running, Pending, etc.

**Networking**

1. **What is a ClusterIP Service?**

* **Answer:** A ClusterIP service exposes a service on a cluster-internal IP. It is the default type of service and allows communication between pods within the cluster.

1. **What is a NodePort Service?**

* **Answer:** A NodePort service exposes a service on a specific port across all nodes. It enables external access to a service by connecting to any node’s IP address and port.

1. **What is a LoadBalancer Service?**

* **Answer:** A LoadBalancer service automatically provisions an external load balancer (typically cloud-based) to expose a service to the external world.

1. **What is an Ingress in Kubernetes?**

* **Answer:** An Ingress is an API object that manages external access to services, typically HTTP or HTTPS. It can provide SSL termination, path-based routing, and load balancing.

1. **How does Kubernetes provide Service Discovery?**

* **Answer:** Kubernetes provides DNS-based service discovery. Each service is automatically assigned a DNS name, and pods can use this name to discover services.

1. **What are Network Policies?**

* **Answer:** Network Policies are rules that control the traffic flow between pods. They define what traffic is allowed and blocked, enhancing security and isolation.

**Storage**

1. **What is a PersistentVolume (PV)?**

* **Answer:** A PersistentVolume is a piece of storage in the Kubernetes cluster that is provisioned by an administrator. It represents a resource in the cluster that can be used by pods.

1. **What is a PersistentVolumeClaim (PVC)?**

* **Answer:** A PersistentVolumeClaim is a request for storage by a user. It binds to an available PersistentVolume and allows a pod to use it.

1. **What is the difference between a Volume and a PersistentVolume in Kubernetes?**

* **Answer:** A Volume is a storage resource that exists for the duration of a pod, while a PersistentVolume is an independent storage resource that can be reused across different pods.

1. **What is a StatefulSet in Kubernetes?**

* **Answer:** A StatefulSet manages stateful applications by ensuring that each pod has a stable identity and persistent storage. It is useful for applications like databases.

1. **What is a ConfigMap?**

* **Answer:** A ConfigMap is an object used to store non-sensitive configuration data in key-value pairs. It can be used by pods to dynamically configure applications.

1. **What is a Secret in Kubernetes?**

* **Answer:** A Secret is an object used to store sensitive data like passwords, tokens, and keys. It is encrypted and can be mounted as environment variables or files.

**Security**

1. **What are Taints and Tolerations?**

* **Answer:** Taints allow nodes to repel certain pods, while tolerations allow pods to be scheduled on nodes with matching taints. This is useful for isolating workloads.

1. **How can you secure Kubernetes API access?**

* **Answer:** You can secure Kubernetes API access using authentication methods such as RBAC (Role-Based Access Control), certificates, and API tokens.

1. **What is Role-Based Access Control (RBAC)?**

* **Answer:** RBAC is a method of controlling access to Kubernetes resources based on the roles assigned to users or service accounts. Roles define permissions, and bindings link roles to users or services.

1. **What is a ServiceAccount in Kubernetes?**

* **Answer:** A ServiceAccount is an identity used by pods or controllers to access the Kubernetes API and interact with other services within the cluster.

1. **How can you limit resource usage in Kubernetes?**

* **Answer:** Resource usage can be limited using **ResourceRequests** and **ResourceLimits** in Pod specifications, which ensure that pods don’t consume excessive CPU or memory.

1. **What is NetworkPolicy in Kubernetes?**

* **Answer:** A NetworkPolicy defines rules for controlling the traffic flow between pods. It enables network segmentation and isolation for better security.

**Scheduling and Scaling**

1. **What is the role of the Kubernetes Scheduler?**

* **Answer:** The Kubernetes Scheduler assigns pods to nodes in the cluster based on available resources, constraints, and affinity rules.

1. **What is Horizontal Pod Autoscaler (HPA)?**

* **Answer:** The HPA automatically adjusts the number of pod replicas in a deployment based on CPU utilization or custom metrics.

1. **What is Vertical Pod Autoscaler (VPA)?**

* **Answer:** VPA automatically adjusts the CPU and memory resource requests and limits for pods based on usage.

1. **What is a DaemonSet in Kubernetes?**

* **Answer:** A DaemonSet ensures that a copy of a pod is running on each node in the cluster. It is useful for background tasks like monitoring or logging.

1. **How can you scale a deployment in Kubernetes?**

* **Answer:** You can scale a deployment by running the command kubectl scale deployment <deployment-name> --replicas=<num-of-replicas> or by modifying the replica count in the deployment YAML.

1. **What is the difference between Horizontal and Vertical Autoscaling?**

* **Answer:** Horizontal scaling involves adding more pods, while vertical scaling involves increasing the resources (CPU/memory) for existing pods.

**Troubleshooting**

1. **How can you debug a failing pod in Kubernetes?**

* **Answer:** You can debug a failing pod using commands like kubectl logs <pod-name> to view logs, kubectl describe pod <pod-name> to get detailed information, and kubectl exec to run commands inside a container.

1. **How can you view the logs of a pod?**

* **Answer:** Use the command kubectl logs <pod-name> to view the logs. If there are multiple containers, specify the container name as well.

1. **How can you get detailed information about a specific pod?**

* **Answer:** Use kubectl describe pod <pod-name> to get detailed information about the pod, including events, status, and resource usage.

1. **What are the common causes of pod failures?**

* **Answer:** Common causes of pod failures include insufficient resources (CPU/memory), incorrect configuration, image pull errors, or unavailability of dependent services.

1. **What is the purpose of kubectl get events?**

* **Answer:** kubectl get events displays the events in a cluster, which can help identify issues such as pod scheduling failures, crashes, or node problems.

1. **How do you check the status of a node in Kubernetes?**

* **Answer:** Use the command kubectl get nodes to view the status of nodes. For more details, you can use kubectl describe node <node-name>.

**Advanced Topics**

1. **What is Helm in Kubernetes?**

* **Answer:** Helm is a package manager for Kubernetes that simplifies deploying and managing applications by using Helm charts.

1. **What is a Kubernetes Operator?**

* **Answer:** An Operator is a Kubernetes controller that manages complex, stateful applications by extending the Kubernetes API to automate tasks like backups or scaling.

1. **What is the role of the Kubelet?**

* **Answer:** Kubelet is an agent that runs on each node. It ensures that containers are running in a pod and manages their lifecycle.

1. **What is the difference between StatefulSet and Deployment?**

* **Answer:** A StatefulSet is used for stateful applications and ensures stable storage and network identity. A Deployment is for stateless applications and does not guarantee stable identifiers.

1. **What is the role of Kube Proxy in Kubernetes?**

* **Answer:** Kube Proxy manages network traffic to services and ensures that traffic is correctly routed to pods. It provides load balancing for services.

**Advanced Topics**

1. **What are Helm Charts in Kubernetes?**

* **Answer:** Helm Charts are packages of pre-configured Kubernetes resources that can be deployed easily. They are templates that allow the configuration and management of applications within Kubernetes.

1. **What is the difference between Helm 2 and Helm 3?**

* **Answer:** Helm 3 is a major upgrade from Helm 2. The key differences include:
  + Helm 3 no longer uses Tiller (the server-side component).
  + Helm 3 has better security and improved release management.

1. **What is the function of the Kubernetes API Server?**

* **Answer:** The API Server is the central component that interacts with the Kubernetes cluster's control plane. It exposes the Kubernetes API, processes requests, and serves as the gateway for interacting with Kubernetes resources.

1. **What is a CronJob in Kubernetes?**

* **Answer:** A CronJob allows you to run jobs at scheduled times, similar to cron jobs in Linux. These jobs are useful for tasks that need to be performed periodically.

1. **What is the difference between a Job and a CronJob in Kubernetes?**

* **Answer:** A Job is used to run a pod until completion, while a CronJob runs jobs on a scheduled basis, similar to cron in Unix/Linux systems.

1. **What is the function of the Kubernetes Controller Manager?**

* **Answer:** The Controller Manager is responsible for managing controllers in Kubernetes. Controllers handle tasks like maintaining the desired state for ReplicaSets, Deployments, and other resources.

1. **What is the role of etcd in Kubernetes?**

* **Answer:** etcd is a distributed key-value store used by Kubernetes to store all configuration data and the state of the cluster, including details of nodes, pods, and deployments.

1. **What is the role of the Kubernetes Scheduler?**

* **Answer:** The Scheduler is responsible for selecting which node an unscheduled pod will run on based on the available resources and constraints.

1. **What is an Affinity in Kubernetes?**

* **Answer:** Affinity rules are used to control which nodes a pod can be scheduled on based on labels. There are two types of affinity: node affinity and pod affinity.

1. **What is Anti-Affinity in Kubernetes?**

* **Answer:** Anti-affinity rules allow you to specify that certain pods should not run on the same node or in a certain zone, helping to avoid resource contention or failure.

**Kubernetes Scheduling**

1. **What is Pod Affinity?**

* **Answer:** Pod Affinity allows you to constrain a pod to be scheduled on nodes where specific pods are running, helping with co-location and performance optimization.

1. **What is a DaemonSet used for?**

* **Answer:** A DaemonSet ensures that a copy of a pod runs on every node in the cluster. It is typically used for cluster-wide services like logging, monitoring, or network proxies.

1. **What is the difference between StatefulSet and DaemonSet?**

* **Answer:** A StatefulSet is used for managing stateful applications, while a DaemonSet ensures that a pod runs on each node in the cluster for background tasks.

1. **How does Kubernetes handle resource allocation and limits?**

* **Answer:** Kubernetes allows you to specify resource requests and limits for CPU and memory in the Pod's configuration. Requests are the resources a pod is guaranteed to have, while limits are the maximum resources it can use.

1. **What is the function of the Kubelet?**

* **Answer:** The Kubelet is an agent that runs on each node and ensures that containers are running in the pods. It also reports back the status of the node and pod to the API server.

1. **What is the function of Kube Proxy?**

* **Answer:** Kube Proxy manages network traffic inside the Kubernetes cluster. It routes the traffic to the correct pods, providing load balancing and service discovery.

1. **How does Kubernetes provide High Availability (HA)?**

* **Answer:** Kubernetes provides High Availability by replicating control plane components (API server, etcd, scheduler) across multiple nodes and maintaining multiple replicas of application pods.

1. **What is the role of a ReplicaSet in Kubernetes?**

* **Answer:** A ReplicaSet ensures that a specified number of pod replicas are running at any given time. It automatically replaces any pods that fail or are deleted.

1. **What are the main differences between Kubernetes Deployments and ReplicaSets?**

* **Answer:** A Deployment manages the creation and updates to ReplicaSets, while a ReplicaSet ensures a specific number of pod replicas are running. Deployments provide features like rolling updates.

1. **How does Kubernetes manage updates and rollbacks of applications?**

* **Answer:** Kubernetes handles rolling updates through Deployments, allowing pods to be updated incrementally. If an update fails, you can roll back to the previous version.

**Storage and Volumes**

1. **What is a Persistent Volume (PV)?**

* **Answer:** A Persistent Volume (PV) is a piece of storage in the Kubernetes cluster that has been provisioned by an administrator. It is independent of the lifecycle of pods and can be reused.

1. **What is the difference between a PersistentVolume and a PersistentVolumeClaim (PVC)?**

* **Answer:** A PersistentVolume is a storage resource, while a PersistentVolumeClaim is a request for storage. PVCs bind to PVs, which then provide storage for pods.

1. **What is a StorageClass in Kubernetes?**

* **Answer:** A StorageClass provides a way to define different types of storage and how they should be provisioned. It allows users to dynamically provision Persistent Volumes with different parameters.

1. **What is an NFS volume in Kubernetes?**

* **Answer:** An NFS volume allows you to mount a directory from an NFS server into your Kubernetes pods. It is useful for sharing storage between multiple pods.

1. **What are the different types of Volumes available in Kubernetes?**

* **Answer:** Types of volumes in Kubernetes include **emptyDir, hostPath, NFS, persistentVolume, configMap, secret,** and more, each serving a different purpose.

**Security and RBAC**

1. **What is Role-Based Access Control (RBAC)?**

* **Answer:** RBAC in Kubernetes is a policy mechanism that regulates access to resources based on the roles assigned to users, groups, or service accounts.

1. **How do you secure communication between Pods in Kubernetes?**

* **Answer:** You can use Network Policies to secure communication between pods and restrict traffic flow. Additionally, you can implement encryption for inter-pod communication.

1. **What is Kubernetes ServiceAccount?**

* **Answer:** A ServiceAccount is a special type of account used by pods and controllers to access the Kubernetes API and interact with other services in the cluster.

1. **What is the purpose of Kubernetes Secrets?**

* **Answer:** Kubernetes Secrets are used to store sensitive information like passwords, tokens, and SSH keys. They are securely stored and can be used by pods.

1. **What is the difference between a Secret and a ConfigMap?**

* **Answer:** A ConfigMap is used for storing non-sensitive data in key-value pairs, while a Secret stores sensitive information. Secrets are encoded and can be encrypted for added security.

**Monitoring and Logging**

1. **How can you monitor a Kubernetes cluster?**

* **Answer:** Kubernetes can be monitored using tools like Prometheus, Grafana, and Kubernetes Dashboard, which collect and visualize metrics from the cluster.

1. **What is the role of Kubernetes Dashboard?**

* **Answer:** The Kubernetes Dashboard provides a web-based user interface to manage and monitor cluster resources, view pod statuses, and manage deployments.

1. **How does Kubernetes handle logging?**

* **Answer:** Kubernetes provides centralized logging through tools like Fluentd, Elasticsearch, and Kibana. Logs from containers are collected and stored in centralized locations for analysis.

1. **How can you troubleshoot pod issues in Kubernetes?**

* **Answer:** Use the following commands to troubleshoot pod issues:
  + kubectl logs <pod-name>
  + kubectl describe pod <pod-name>
  + kubectl get events
  + kubectl exec (to run commands inside the pod).

1. **What are Kubernetes Metrics Server and Prometheus used for?**

* **Answer:** The Metrics Server collects resource usage metrics (CPU and memory) from nodes and pods. Prometheus is a monitoring system that collects and stores metrics for more detailed analysis.

**Kubernetes Updates and Maintenance**

1. **How does Kubernetes handle upgrades?**

* **Answer:** Kubernetes handles upgrades by updating the master and worker nodes in a rolling fashion to avoid downtime. The kubeadm tool is often used for upgrading clusters.

1. **What is the process for scaling up a Kubernetes cluster?**

* **Answer:** To scale up a Kubernetes cluster, you add more nodes and ensure the control plane and worker nodes are properly configured to handle the additional load.

1. **How does Kubernetes handle node failures?**

* **Answer:** Kubernetes ensures high availability by rescheduling pods from failed nodes to healthy nodes, keeping the cluster’s desired state intact.

1. **What is a rolling update in Kubernetes?**

* **Answer:** A rolling update gradually replaces pods with new ones to ensure zero downtime during application updates.

1. **What is the rollback feature in Kubernetes?**

* **Answer:** The rollback feature allows you to revert to a previous version of a Deployment if the current version fails.

**General Kubernetes Knowledge**

1. **What is a Kubernetes Operator?**

* **Answer:** A Kubernetes Operator is a custom controller that manages the lifecycle of complex applications and automates tasks such as scaling, backups, and application-specific management.

1. **What are Custom Resource Definitions (CRDs)?**

* **Answer:** CRDs allow you to extend the Kubernetes API by creating custom resources. This enables the creation of custom objects specific to an application’s needs.

1. **What is Kubernetes Federation?**

* **Answer:** Federation allows you to manage multiple Kubernetes clusters as one, enabling cross-cluster replication and scaling.

1. **What is a Pod Disruption Budget (PDB)?**

* **Answer:** A PDB allows you to specify the minimum number of replicas of a pod that must be available during voluntary disruptions like draining nodes for maintenance.

1. **What is Kubernetes' approach to CI/CD?**

* **Answer:** Kubernetes supports continuous integration and continuous deployment (CI/CD) pipelines by integrating with tools like Jenkins, GitLab CI, and Argo CD to automate the application lifecycle.

1. **What is a Kubernetes StatefulSet used for?**

* **Answer:** A StatefulSet is used for managing stateful applications that require stable, unique network identifiers and persistent storage.

1. **How does Kubernetes handle resource limits?**

* **Answer:** Kubernetes allows you to define **resource requests** and **limits** for pods. Requests specify the minimum resources needed, while limits specify the maximum resources a pod can use.

1. **What is a Kubernetes ServiceAccount?**

* **Answer:** A ServiceAccount provides an identity for processes that run in a pod, allowing them to interact with the Kubernetes API server.

1. **How does Kubernetes manage workload distribution across nodes?**

* **Answer:** Kubernetes uses the scheduler to manage workload distribution, taking into account node resources, affinity rules, and constraints.

1. **What is the purpose of Kubernetes namespaces?**

* **Answer:** Namespaces provide a way to divide cluster resources between different users or teams, providing isolation and better resource management.