Kubernetes basic

Kubernetes, also known as K8s, is an open-source platform designed to automate deploying, scaling, and operating application containers. Here's a basic overview of Kubernetes:

What is Kubernetes?

- Kubernetes is a system for running and coordinating containerized applications across a cluster of machines. It is aimed at providing predictability, scalability, and high availability for your applications.

Key Features of Kubernetes:

- **Container Orchestration:** Kubernetes allows users to deploy and manage containers across a cluster of machines. It abstracts away the underlying infrastructure, providing a consistent environment for running applications.
- **Automated Scaling:** Kubernetes automatically scales applications based on resource utilization or user-defined metrics. It can scale applications horizontally by adding or removing instances as needed.
- **Self-Healing:** Kubernetes monitors the health of applications and automatically restarts containers that fail or become unresponsive. It also replaces failed nodes and reschedules pods to ensure high availability.
- **Service Discovery and Load Balancing:** Kubernetes provides built-in service discovery and load balancing for containerized applications. Services abstract away individual pods and enable communication between different components of an application.
- **Storage Orchestration:** Kubernetes offers storage orchestration capabilities, allowing users to dynamically provision and manage storage volumes for applications. It supports various storage solutions, including local storage, network-attached storage (NAS), and cloud storage.
- **Configuration Management:** Kubernetes allows users to define configuration settings for applications using ConfigMaps and Secrets. This enables the separation of configuration from application code and facilitates environment-specific configurations.

Jay Bhadreshwara

How Kubernetes Works:

- **Cluster Setup:** Kubernetes clusters consist of one or more master nodes and multiple worker nodes. The master node manages the cluster's control plane, while worker nodes host the containerized applications.
- **Pods:** The smallest deployable unit in Kubernetes is a pod, which encapsulates one or more containers. Pods share networking and storage resources and are scheduled and managed as a single unit.
- **Deployments:** Deployments define desired state for applications, including the number of replicas and container images. Kubernetes ensures that the current state matches the desired state by creating, updating, or deleting pods as necessary.
- **Services:** Services provide network connectivity to pods, enabling communication between different parts of an application. Kubernetes offers various types of services, including ClusterIP, NodePort, and LoadBalancer, to expose applications internally or externally.
- **Controllers:** Kubernetes controllers continuously monitor the cluster's state and reconcile it with the desired state defined in the Kubernetes objects. Examples of controllers include ReplicaSet, Deployment, StatefulSet, and DaemonSet.
- **API Server:** The Kubernetes API server exposes a RESTful interface for interacting with the cluster. Users and automated systems can use kubectl or other client libraries to perform operations such as deploying applications, scaling resources, and inspecting cluster state.

Overall, Kubernetes provides a powerful platform for deploying and managing containerized applications at scale, enabling organizations to achieve higher agility, scalability, and reliability in their infrastructure.