Kubernetes

Kubernetes, often abbreviated as K8s, is an open-source container orchestration platform designed to automate the deployment, scaling, and management of containerized applications.

Containerization:

Kubernetes leverages container technology (like Docker) to encapsulate applications and their dependencies, ensuring consistency across different environments.

Key Components

1. **Node** : A physical or virtual machine where containers run.

2. **Pod** : The smallest deployable unit in Kubernetes, representing one or more containers sharing the same network and storage.

3. **Service** : An abstraction that defines a logical set of Pods and a policy by which to access them.

4. **Deployment** : A higher-level abstraction that manages the deployment and scaling of a set of Pods.

5. **Control Plane** : Consists of components that manage and control the state of the cluster. Includes the API Server, Controller Manager, etcd, and Scheduler.

6. **Worker Nodes** : Machines (physical or virtual) responsible for running containers. Each node runs a container runtime (e.g., Docker) and a kubelet to communicate with the control plane.

7. **Scaling** : Kubernetes can scale applications automatically based on demand or defined policies.
   Horizontal Pod Autoscaler (HPA) is used to adjust the number of running Pods.

8. **Service Discovery and Load Balancing** :
   Kubernetes provides DNS for service discovery and a built-in load balancer to distribute traffic across multiple Pods.

9. **Rolling Updates and Rollbacks** :
   Deployments in Kubernetes support rolling updates, allowing new versions of an application to be gradually rolled out while monitoring the health of the Pods. Rollbacks can be performed in case of issues.

10. **Configuration Management** : Kubernetes allows you to store and manage configuration separately from application code using ConfigMaps and Secrets.

11. **Storage Orchestration** :
    Provides mechanisms for containerized applications to use storage resources, including dynamic provisioning and mounting volumes.

12. **Self-Healing** :
    Kubernetes monitors the state of applications and automatically restarts or replaces failed containers.

13. **Portability and Multi-Cloud** :
    Kubernetes provides a consistent environment across various infrastructure providers, promoting multi-cloud and hybrid cloud deployments.

14. **Security** :
    Implements role-based access control (RBAC), network policies, and secrets management to enhance security.

15. **Extensibility** :
    Kubernetes is highly extensible, allowing the addition of custom resources and controllers.

16. **Community and Ecosystem** :
    Kubernetes has a large and active community, contributing to a vast ecosystem of tools and extensions.