NOTES

#Kubernetes In 30Days challenge :-

#Day29 :-

Kubernetes Interview Questions.

1) You need to deploy a microservice architecture with Kubernetes. How would you design the service discovery mechanism and what tools would you use.

sol :- I would Implement service discovery using Kubernetes services. Tools like CoreDNS or external solutions like Consul can enhance service discovery capabilities. creating labels and selectors for services and utilizing Ingress controllers or API gateways are additional strategies for managing service discovery in microservice architecture.

2) Your team is adopting GitOps practices for managing k8s configurations. Explain the GitOps workflow and benefits it provides.

sol :- GitOps is a declarative approach where

the entire kubernetes configuration is stored in a Git repository. changes to the cluster are made through pull requests to the repo. A GitOps tool (ArgoCD, Flux) continously monitors the repository and reconciles the actual state with the desired state in the Git. Benefits include versioning, auditability and ability rollback changes easily.

3) What is Pod Distribution Budget in kubernetes.

sol:- Pod Distribution Budget (PDB) is used to limit the no. of concurrently disrupted pods during voluntarily disruptions such as draining a node for maintainance. It helps ensure high availability by avoiding scenarios where too many pods are terminated simultaneously, impacting the application stability. PDB is crucial in scenarios where minimizing downtime is priority.

4) Your team is deploying a stateful application with the high data consistency requirements how would you configure storage, and handle data persistance in a kubernetes statefulset

Sol:- In a statefulset, you would configure persistent volumes(PVs) and Persistent Volume claims(PVC) to ensure data persistence. Using storage classes, volume claims can be dynamically provisioned and the data can be stored in network-attached storage or other persistent storage solutions. Its crucial to carefully manage pod termination and startup sequences to maintain data consistency.

Thanks for Reading!

Bye!