

# Orquestração de Contêiner

Kubernetes

# O que é Kubernetes?

- Kubernetes, abreviado como K8s, é uma ferramenta de orquestração de contêineres de código aberto.
- Automação de implantação, escalonamento e operação de aplicativos containerizados.

# História do Kubernetes

- Desenvolvido a partir da experiência do Google com o Borg.
- Projetado para crescer de dezenas a milhões de contêineres.

# Benefícios do Kubernetes

- Velocidade e Recuperação:
  - Aumento de velocidade na liberação e recuperação de aplicações.
  - Mecanismos de autocura.

# Estrutura do Kubernetes

- Sistema Distribuído:
  - Várias máquinas formam um cluster.
  - Pode incluir máquinas físicas e virtuais, on-premise ou na nuvem.

# Algoritmos de Agendamento

- Agendamento Inteligente:
  - Considera recursos disponíveis, prioridade e outras restrições.
  - Move contêineres conforme necessário.

# Compatibilidade com Runtimes

- Agnóstico em relação ao Runtime:
  - Suporta diferentes runtimes de contêineres (Docker, Rocket).

# Adaptação e Adoção

- Design Modular:
  - Facilitou a adoção ampla.
  - Configuração declarativa para deploy rápido e fácil.



# Recursos para Equipes de Operação

- Autocura e Manutenção:
  - Move contêineres de máquinas com falhas.
  - Recursos integrados para manutenção.

# Federação de Clusters

- Redundância:
  - Múltiplos clusters podem se unir.
  - Contêineres se movem automaticamente entre clusters.

# Recursos Destacados do Kubernetes

- Principais Recursos:
  - Rollout e rollback automáticos.
  - Escalonamento horizontal sem interrupções.
  - Gerenciamento de segredos.
  - Descoberta de serviços e balanceamento de carga.
  - Suporte para contêineres Linux e Windows.
  - Coleta de logs.
  - Suporte a aplicativos stateful.
  - Gerenciamento de volumes persistentes.
  - Cotas de CPU e memória.
  - Processamento de jobs em batch.
  - Controle de acesso baseado em funções.

# Comparação com Outras Ferramentas

- Ferramentas Comparadas:
  - DCOS (Distributed Cloud Operating System)
  - Amazon ECS (Elastic Container Service)
  - Docker Swarm Mode

# DCOS

- Descrição e Benefícios:
  - Agrupa recursos de computação em um pool uniforme.
  - Suporte a diferentes tipos de cargas de trabalho.
  - Inclui um Gerenciador de Pacotes.

# Amazon ECS

- Descrição e Benefícios:
  - Orquestração de contêineres dentro da AWS.
  - Pools de recursos de computação gerenciados via API.
  - Menos recursos em comparação com ferramentas de código aberto.

# Docker Swarm Mode

- Descrição e Benefícios:
  - Solução oficial do Docker para orquestração de contêineres.
  - Conjunto de recursos semelhante ao Kubernetes e DCOS.
  - Integração nativa com o comando docker.

# Clusters Kubernetes c/ um Nó

- Depois de decidir usar o Kubernetes, você tem várias opções para implantá-lo. Esta lição discute algumas dessas opções.



# Cluster de Nó Único

- Implantação em Nó Único:
  - Ideal para desenvolvimento e teste.
  - Docker para Mac e Docker para Windows suportam Kubernetes localmente.
  - Certifique-se de que o Kubernetes está habilitado nas configurações.

# Minikube e Kubeadm

- Outras Opções:
  - Minikube: Suporta Linux, Mac e Windows.
  - Kubeadm: Configura clusters de nó único em sistemas Linux, instala o Kubernetes no sistema.

# Uso em Integração Contínua

- Clusters de Nó Único em CI:
  - Criar clusters efêmeros rapidamente.
  - Kubernetes em Docker (K-in-D ou kind) para pipelines de integração contínua.

# Cluster de Múltiplos Nós

- Implantação em Múltiplos Nós:
  - Ideal para cargas de trabalho de produção.
  - Aproveita o escalonamento horizontal e a tolerância a falhas.

# Controle vs. Manutenção

- Questões Importantes:
  - Quanto controle você quer sobre o cluster?
  - Quanto esforço está disposto a investir na manutenção?

# Soluções Totalmente Gerenciadas

- Soluções Gerenciadas:
  - Liberam você da manutenção rotineira.
  - Podem atrasar algumas versões do Kubernetes.
  - Exemplos: Amazon EKS, Azure AKS, Google GKE.

# Soluções de Controle Total

- Controle Total:
  - Ferramentas: kubespray, kops, kubeadm.
  - Pergunte-se sobre o investimento e a expertise em um provedor de nuvem específico.

# Suporte Empresarial

- Suporte Empresarial:
  - Ofertas adicionais e suporte de fornecedores.
  - Exemplos: OpenShift, Pivotal Container Service, Rancher.



# Dependência de Fornecedor

- Evitar Dependência de Fornecedor:
  - Foco em soluções de código aberto.
  - Ferramentas: kubespray, Rancher.

# Localização do Cluster

- Cluster On-Prem, Nuvem ou Ambos:
  - Kubernetes oferece abstração de cluster de recursos.
  - Suporta diferentes plataformas.
  - Exemplos: GKE on-prem, EKS com nós on-premise, Azure Stack.

# Suporte a Contêineres

- Tipos de Contêineres:
  - Contêineres Linux: Nós Linux no cluster.
  - Contêineres Windows: Nós Windows no cluster.
  - Suporte para ambos os tipos de contêineres no mesmo cluster.