



Curso Bônus

Planejando Sua Carreira Para as
Profissões do Futuro

Kubernetes x Docker Swarm



Monitorar e gerenciar ciclos de vida de containers em ambientes mais complexos requer uma ferramenta de orquestração de containers.

Embora o Docker inclua sua própria ferramenta de orquestração (chamada Docker Swarm), a maioria dos profissionais que usam Docker escolhe o Kubernetes.

Kubernetes é uma plataforma de orquestração de containers de código aberto descendente de um projeto desenvolvido para uso interno no Google.

O Kubernetes agenda e automatiza tarefas de gerenciamento de arquiteturas baseadas em container, incluindo implantação de containers, atualizações, descoberta de serviço, provisionamento de armazenamento, balanceamento de carga, monitoramento de integridade e muito mais.

Além disso, o ecossistema de ferramentas de código aberto para Kubernetes - que inclui Istio e Knative - permite que as organizações implantem uma plataforma como serviço (PaaS) de alta produtividade para aplicativos em container e um acesso mais rápido à computação sem servidor (serverless).

A orquestração de containers é vital no ambiente de tecnologia moderno, onde automatizar o gerenciamento de microsserviços de aplicativos é mais importante do que nunca. E isso vem se tornando o padrão, em especial em projetos de Engenharia de Dados.

Mas qual a alternativa ideal para a orquestração de containers? Há um forte debate sobre a ferramenta para esta atividade, o Docker Swarm ou o Kubernetes.

Vamos explorar as semelhanças e diferenças entre o Docker Swarm e o Kubernetes.

O Que São Containers?

Em poucas palavras, os containers são uma maneira padrão de empacotar aplicativos e todas as suas dependências para que você possa mover os aplicativos sem problemas entre diferentes ambientes e isolar o funcionamento de diferentes módulos de uma aplicação maior. Ao empacotar o código, as dependências e as configurações de um aplicativo em um bloco de construção fácil de usar, os containers permitem que você tome medidas importantes para reduzir o tempo de implantação e melhorar a confiabilidade do aplicativo.

Em aplicativos corporativos, o número de containers pode crescer rapidamente para um número incontrolável. Para usar os containers com mais eficiência, você precisará orquestrar os aplicativos em container, que é onde o Kubernetes e o Docker Swarm entram em ação.

Kubernetes

O Kubernetes é uma plataforma de código aberto para gerenciar containers, as cargas de trabalho complexas e escalabilidade. Com o Kubernetes, desenvolvedores e equipes de DevOps podem agendar, implantar e gerenciar aplicativos altamente disponíveis usando a flexibilidade dos clusters. Um cluster Kubernetes é composto de hosts de computação chamados de nós de trabalho. Esses nós são gerenciados por um mestre (master) do cluster Kubernetes que controla e monitora todos os recursos no cluster. Um nó pode ser uma máquina virtual (VM) ou uma máquina física.

Vantagens do Kubernetes:

- O Kubernetes oferece uma ampla variedade de funcionalidades importantes, incluindo descoberta de serviços, entrada e balanceamento de carga, auto

recuperação, orquestração de armazenamento, escalabilidade horizontal, rollouts/rollbacks automatizados e execução em lote.

- Ele tem um conjunto unificado de APIs e fortes garantias sobre o estado do cluster.
- Tem uma comunidade de código aberto que é muito ativa no desenvolvimento da base de código.
- O Kubernetes tem a maior adoção do mercado.
- É testado por grandes players como o Google e IBM, e é executado na maioria dos sistemas operacionais.
- Está disponível na nuvem ou para uso local e possui ofertas gerenciadas ou não gerenciadas de todos os grandes provedores de nuvem (AWS, Microsoft Azure, Google Cloud Platform, etc.).

Docker Swarm

O Docker Swarm é outra plataforma de orquestração de containers de código aberto que existe há algum tempo. Swarm — ou mais precisamente, modo swarm — é o suporte nativo do Docker para orquestrar clusters de containers Docker.

Um cluster Swarm consiste em nós do gerenciador Swarm implantados pelo Docker Engine (que orquestram e gerenciam o cluster) e nós do trabalhador (que são direcionados para executar tarefas pelos nós do gerenciador).

Vantagens do Docker Swarm:

- O Docker é uma plataforma usada para criar e implantar aplicativos em container. O Swarm foi desenvolvido para uso com o Docker Engine.
- É fácil de instalar e configurar para um ambiente Docker.
- Ferramentas, serviços e software executados com containers Docker também funcionarão bem com o Swarm.
- Ele tem sua própria API Swarm.
- Ele se integra perfeitamente às ferramentas do Docker, como Docker Compose e Docker CLI, pois usa a mesma interface de linha de comando (CLI) que o Docker Engine.
- Ele usa um sistema de filtragem e agendamento para fornecer seleção inteligente de nós, permitindo que você escolha os nós ideais em um cluster para implantação de containers.

Kubernetes x Docker Swarm

Ambas as plataformas permitem que você gerencie containers e dimensione a implantação de aplicativos. Suas diferenças são uma questão de complexidade.

O Kubernetes oferece um meio eficiente de gerenciamento de containers que é ótimo para aplicativos de alta demanda com configuração complexa, enquanto o Docker Swarm foi projetado para facilitar o uso, tornando-o uma boa opção para aplicativos simples que são rápidos de implantar e fáceis de gerenciar.

Aqui estão algumas diferenças detalhadas entre o Docker Swarm e o Kubernetes:

Instalação e configuração. Devido à complexidade do Kubernetes, o Docker Swarm é mais fácil de instalar e configurar.

- **Kubernetes:** a instalação manual pode ser diferente para cada sistema operacional. Nenhuma instalação é necessária para ofertas gerenciadas de provedores de nuvem.
- **Swarm:** há uma instalação simples com o Docker e as instâncias geralmente são consistentes em todos os sistemas operacionais.

Escalabilidade. O Kubernetes oferece dimensionamento completo com base no tráfego, enquanto o Docker Swarm enfatiza o dimensionamento rápido.

- **Kubernetes:** o escalonamento automático horizontal está integrado.
- **Swarm:** oferece escalonamento automático de grupos sob demanda.

Balanceamento de carga. O Docker Swarm possui balanceamento de carga automático, enquanto o Kubernetes não. No entanto, um balanceador de carga externo pode ser facilmente integrado por meio de ferramentas de terceiros no Kubernetes.

- **Kubernetes:** a descoberta de serviços é habilitada por meio de um único nome DNS. O Kubernetes tem acesso a aplicativos de container por meio de um endereço IP ou rota HTTP.
- **Swarm:** vem com balanceadores de carga internos.

Alta disponibilidade. Ambas as ferramentas fornecem um alto nível de disponibilidade.

Cursos de Aperfeiçoamento Profissional - Bônus da Formação

- **Kubernetes:** ao desviar o tráfego de nós não saudáveis, o Kubernetes é auto recuperável. Oferece agendamento inteligente e alta disponibilidade de serviços por meio de replicação.
- **Swarm:** os Swarm Managers oferecem controles de disponibilidade e os microsserviços podem ser facilmente duplicados.

Qual Ferramenta de Orquestração de Containers é Ideal?

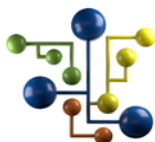
Como a maioria das decisões de plataforma, a ferramenta certa depende das necessidades de cada empresa. Não existe ferramenta ideal. A ferramenta ideal é a que resolve o problema do cliente.

O Kubernetes tem ampla adoção e uma grande comunidade ao seu lado. Ele é suportado por todos os principais provedores de nuvem e ofertas do tipo *faça você mesmo*, como o Docker Enterprise Edition.

O Kubernetes é mais poderoso, personalizável e flexível, com o custo de uma curva de aprendizado inicial mais acentuada. Requer uma equipe experiente e capaz de executá-lo; no entanto, as empresas também estão optando por usar um provedor de serviços gerenciados para simplificar as responsabilidades de gerenciamento de atualizações e permitir que se concentrem na criação de aplicativos.

A vantagem do Docker Swarm vem com familiaridade e ênfase na facilidade de uso. Ele é implantado com o Docker Engine e está prontamente disponível em seu ambiente. Como resultado, o Swarm é mais fácil de começar e pode ser ideal para cargas de trabalho menores.

Usaremos as duas ferramentas aqui no curso.



Equipe DSA

Muito Obrigado!
Continue Trilhando Uma Excelente Jornada de Aprendizagem.