

Passo a Passo para Executar um Script Python em um Cluster Kubernetes

Luiza Ignacio Gomes -> RM:97797

Neste guia, vamos detalhar o processo passo a passo para executar um script em Python em um cluster Kubernetes

**Requisitos:**

- Acesso a um cluster Kubernetes.

- Conhecimento básico sobre Kubernetes e Docker.

- Um script em Python que você deseja executar.

**Script Python:**

Script Python que será executado no cluster Kubernetes:

```python

# script.py

import time

def main():

while True:

print("Executando script...")

time.sleep(5)

if \_\_name\_\_ == "\_\_main\_\_":

main()

```

**Preparação do Script Python:**

Certifique-se de que seu script Python está configurado para ser executado em um ambiente distribuído, se necessário. Isso pode incluir a utilização de bibliotecas como `multiprocessing` ou `celery`.

**Dockerização do Script:**

Para executar o script em um cluster Kubernetes, é necessário criar uma imagem Docker. Crie um arquivo `Dockerfile` que defina o ambiente necessário para o script e copie o script para dentro do contêiner.

**Construção e Upload da Imagem Docker:**

Construa a imagem Docker usando o comando `docker build` e faça o upload para um registro de contêiner, como Docker Hub ou um registro privado no Kubernetes.

**Definição do Deployment no Kubernetes:**

Crie um arquivo de manifesto YAML para definir o deployment no Kubernetes. Este arquivo especifica como o contêiner será executado no cluster.

**Aplicação do Deployment no Cluster:**

Aplique o deployment usando o comando `kubectl apply`.

**Contexto: Infraestrutura para Deploy**

Para garantir a transparência e compreensão do ambiente no qual o deploy será realizado, fornecemos detalhes sobre a infraestrutura utilizada:

- Número de Instâncias do Cluster: O deploy será realizado em um cluster Kubernetes (K8s) composto por [número de instâncias] instâncias.

- Nós (Nodes): O cluster é composto por [número de nós] nós distribuídos para garantir redundância e escalabilidade.

- Versão do microk8s: O Kubernetes é provisionado através da plataforma MicroK8s, na versão [versão do MicroK8s].

A seguir, estão as imagens que demonstram a configuração atual do cluster, incluindo a interface do usuário do MicroK8s e informações sobre os nós do cluster:

[Insira aqui as imagens relevantes com seu nome de usuário visível em todas elas.]

**Logs dos Pods e Informações do Deployment:**

Aqui estão os logs dos pods e as informações do deployment relevantes para este processo:

```bash

microk8s kubectl logs <seu\_pod\_nome> -n <seu\_namespace>

kubectl get all -n <seu\_namespace>

```

[Insira os logs dos pods e as informações do deployment aqui.]

**Conclusão:**

Agora seu script Python está sendo executado em um cluster Kubernetes. Você pode dimensionar o número de réplicas conforme necessário e monitorar o desempenho usando ferramentas do Kubernetes.

Referências:

- [Documentação do Kubernetes](https://kubernetes.io/docs/)

- [Documentação do Docker](https://docs.docker.com/)

- [Repositório de Exemplo no GitHub](https://github.com/seu\_usuario/repo-exemplo)