

# Manual de Criação e configuração de um Cluster de Kubernetes na AWS

Abaixo serão apresentados os passos para a construção de um modelo de sistemas baseado nas tecnologias Kubernetes e AWS como Cloud Provider.

Para a construção do cluster de Kubernetes será utilizado uma ferramenta chamada k3s, que tem como o objetivo subir um cluster de kubernetes com todos os componentes necessários para funcionar, só que de forma com que seja possível executar em pequenos workloads.

A documentação disponível para o K3S pode ser encontrada em: <https://k3s.io/>

## 1. Criação dos nós que farão parte do cluster

### Dicas:

- No momento da criação das máquinas, edite a parte de rede para selecionar as subnets e escolha apenas uma zona de disponibilidade (por exemplo: us-east-1a).
  - Crie um par de chaves para acessar os nós via SSH.
  - Utilize sempre o mesmo grupo de segurança para todas as instâncias, evitando configurações adicionais.
- a. Para este passo, será necessário criar três nós (EC2) na AWS. Acesse o AWS EC2 Console e vá para Instances > Launch Instance. Crie uma instância chamada master e escolha a classe t3.medium (possui 4 GB de memória RAM).
  - b. Em seguida, crie duas instâncias leves (pode-se utilizar instâncias do free-tier, como t2.micro) para atuarem como workers. No momento da configuração do Security Group, selecione o grupo criado automaticamente na criação do nó master.

## 2. Editar o grupo de segurança para acesso às máquinas:

- a. Navegue até o caminho AWS EC2 Console (<https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1>), vá para Security Groups e edite o Security Group criado automaticamente durante a criação da instância master.
- b. Em seguida, adicione as seguintes regras de entrada:
  - i. Permitir todo o tráfego para o próprio Security Group (opção disponível no dropdown). Isso garantirá que todas as máquinas do cluster possam se comunicar entre si.
  - ii. Permitir acesso SSH (protocolo TCP, porta 22) para qualquer endereço IPv4 (0.0.0.0/0).

### 3. Instalar o Kubernetes e inicializar o cluster

- a. Acesse a máquina do nó master, usando a própria console da aws ou o protocolo ssh com a chave criada anteriormente, e execute o comando de instalação do kubernetes com o k3s:

```
curl -sL https://get.k3s.io | sh -
```

- b. Após finalizar, o cluster deve estar rodando e funcionando, para conferir, digite o comando abaixo, a saída do comando deve ser a listagem de nós, com somente 1 nó sendo o master

```
kubectl get nodes
```

- c. Em seguida, copie o token para autenticação dos nós com o comando abaixo:

```
cat /var/lib/rancher/k3s/server/node-token
```

- d. Usando o template abaixo, substitua pelos valores do token copiado anteriormente, e pelo IP privado na máquina master.

```
curl -sL https://get.k3s.io | K3S_URL=https://<informar ip aqui>:6443 K3S_TOKEN=<informar token aqui> sh -
```

### 4. Configuração dos workers:

- a. O próximo passo, é acessar as máquinas workers, e executar o comando gerado no passo anterior.

### 5. Configuração dos target groups

*Um Target Group na AWS é usado para gerenciar e rotear tráfego para um conjunto de instâncias, containers ou endereços IP, no caso vamos usar um Elastic Load Balancer (ELB), o Target Group define os destinos para onde um Load Balancer (ALB ou NLB) enviará as requisições.*

- a. **Acesse o Console da AWS**, vá até o serviço **EC2** e, no menu lateral, selecione **Target Groups**.

#### b. Crie um Novo Target Group

- i. Clique em **Create target group**.
- ii. Escolha o **Target type** como **IP addresses**.
- iii. Defina um nome para o grupo.
- iv. Selecione o **protocolo** ( HTTP).
- v. Informe a porta (a porta pode ser vista no nó master com o comando **kubectl get svc -A**, procure pelo serviço do traefik, ele estará em formato 80:<porta aleatório> é essa porta que você deve informar).
- vi. Escolha o **VPC** correspondente aos IPs que serão adicionados.

#### c. Adicione os IPs

- i. Clique em **Register targets**
- ii. Adicione manualmente os endereços IP desejados.

iii. Confirme clicando em **Include as pending below**.

d. **Configure o Health Check**

- i. Configure a verificação de saúde (Health Check) na aba correspondente
- ii. Defina o protocolo HTTP, e informe a porta que é a mesma do passo B item V
- iii. Expanda o menu relacionado a **Configurações avançadas de verificação de integridade** e altere o código de sucesso de 200, para 404

6. **Configurar um Load balancer**

- a. Acesse o Console da AWS, vá até o serviço EC2 e, no menu lateral selecione Load Balancer
- b. Clique em criar
- c. Selecione Application Load Balancer
- d. Informe a Porta 80
- e. Informe o Target Group criado no passo anterior.

7. **Subir a aplicação**

- a. Copiar o conteúdo do arquivo deployment.yaml e criar o arquivo deployment.yaml dentro do nó master, em seguida executar com o comando abaixo:  
**kubectl apply -f deployment.yaml**

8. **Testar a aplicação**

- a. Se tudo for configurado corretamente, você pode acessar o endereço do load balancer no navegador, e ver a página do nginx inicial.