

Kubernetes – Introdução



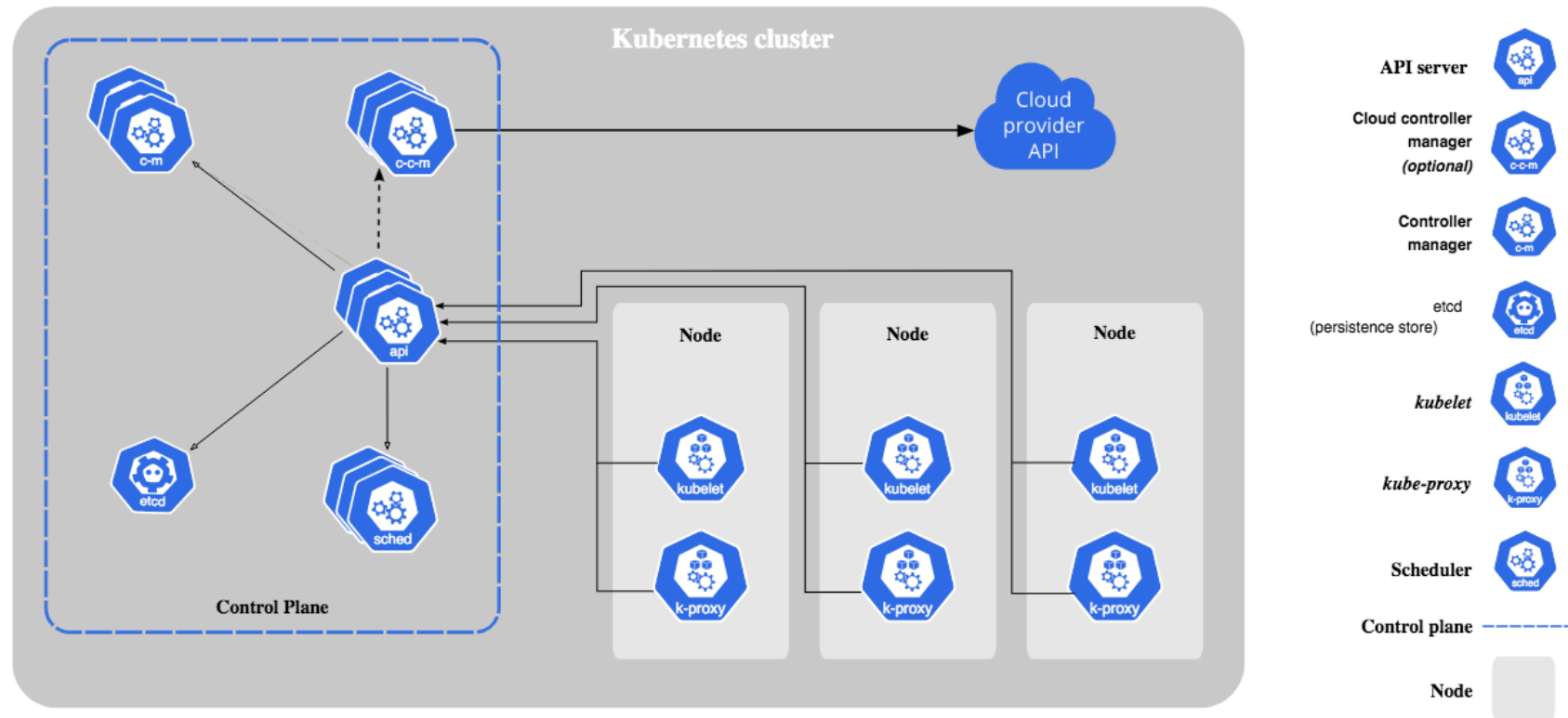
DevOps
Mão na
Massa

O que é e para que serve?

- Plataforma opensource para gerenciamento de containers
- Configuração declarativa de componentes (YAML)
- Organizado em Objects, persistentes no ecossistema do K8s.
- Funcionalidades embarcadas:
 - Load balance
 - Storage
 - Service Discovery
 - Self healing
 - Secrets



Arquitetura



<https://kubernetes.io/docs/concepts/overview/components/>

Arquitetura

1. Nodes

- 1. Kubelet:** Node Agent - Nível dos Pods
- 2. Kube-proxy:** Kubernetes Network proxy (forward, round robin tcp , udp, etc)

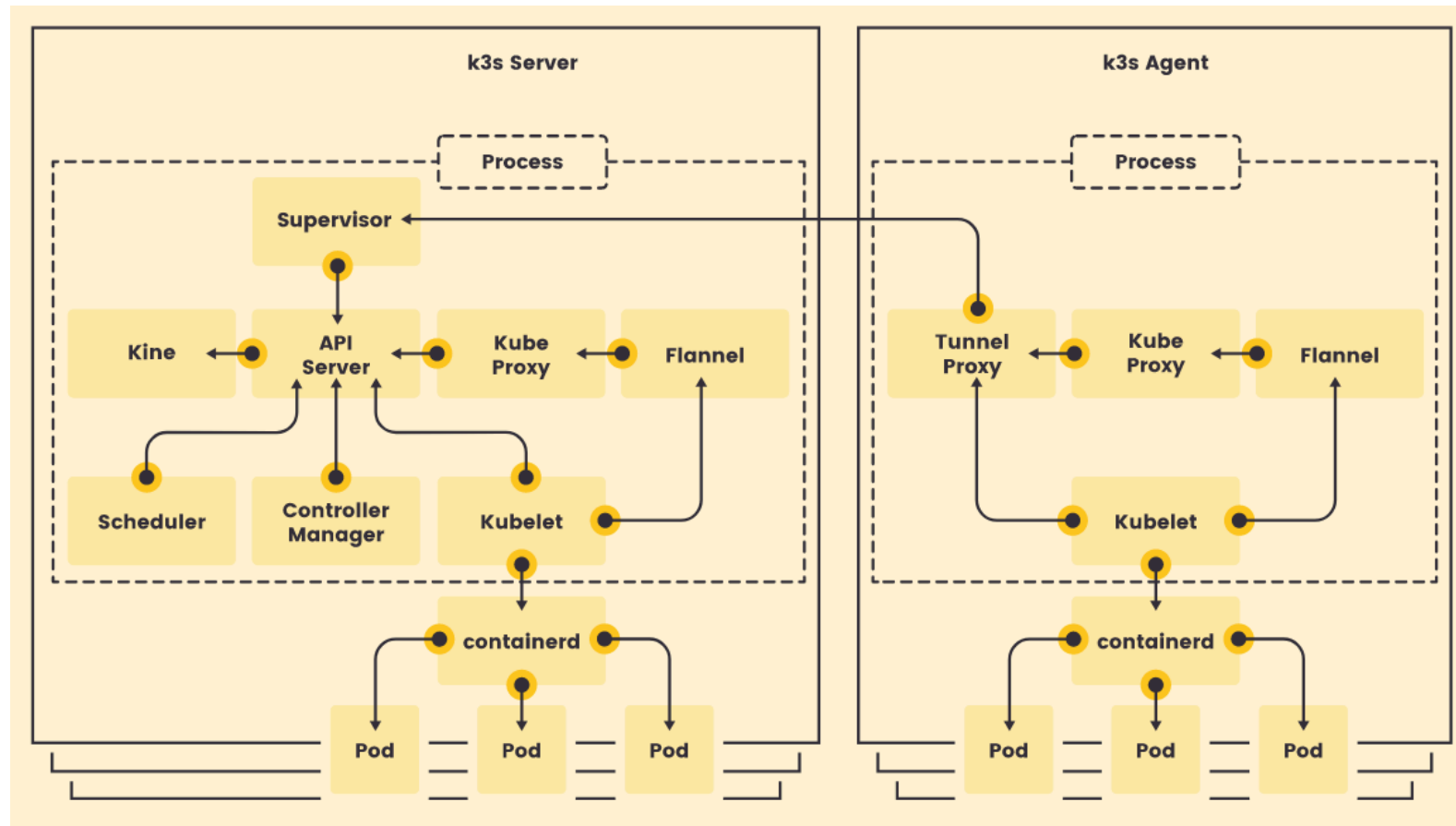
2. Master

- 1. kube-apiserver:** Validação e configuração dos objetos no cluster (pods, services, replication, etc)
- 2. Scheduler:** Camada do control plane que atribui Pods aos Nodes.
- 3. Controllers:** Monitora o estado do cluster, mantendo a consistência do ambiente.

Conceitos

1. **Pod:** É um grupo de um ou mais containers, compartilhando storage e network. Uma especificação de como rodar um container.
2. **Deployment:** Descreve qual é o estado desejado do deploy da aplicação.
3. **kubectl:** Ferramenta de linha de comando que permite controlar o cluster Kubernetes.
4. **Service:** É um mecanismo de exposição das aplicações que rodam nos pods como um serviço de rede.
5. **Ingress:** Exposição do trafego http e https para fora do cluster k8s.
6. **Secrets:** objeto que armazena dados sensíveis, como senhas, chaves, tokens, etc.
7. **ConfigMap:** objeto que armazena chave / valor para dados não sensíveis.

K3s – Distribuição K8s



K3s – mão na massa – Configuração do lab

1. Vagrantfile

```
Vagrant.configure("2") do |config|  
  config.vm.box = 'centos/7'  
  config.vm.hostname = 'manager'  
  config.vm.network "private_network", ip: "192.168.1.2"  
  config.vm.provider 'virtualbox' do |v|  
    v.memory = 2048  
  end  
end
```

Instalar cluster k3s

No manager:

```
curl -sfL https://get.k3s.io | sh -s - --cluster-init --tls-san 192.168.1.2 --node-ip 192.168.1.2 --node-external-ip 192.168.1.2
```


K8s – Configuração do terminal

1. Instalação do Kubens

```
yum install git unzip telnet net-tools -y  
git clone https://github.com/ahmetb/kubectx /opt/kubectx  
sudo ln -s /opt/kubectx/kubectx /usr/local/bin/kubectx  
sudo ln -s /opt/kubectx/kubens /usr/local/bin/kubens
```

1. Instalação do auto complete – bash-completion para kubectl

1. yum install bash-completion -y
2. echo 'source <(kubectl completion bash)' >> ~/.bashrc
3. kubectl completion bash >/etc/bash_completion.d/kubectl
4. echo 'complete -F __start_kubectl k' >> ~/.bashrc

2. Configuracao do Alias k - profile

Configurar registry privado – lab Jenkins

1. Editar arquivo /etc/rancher/k3s/registries.yaml no master node:

```
[root@manager ~]# cat /etc/rancher/k3s/registries.yaml
mirrors:
  docker.io:
    endpoint:
      - "http://192.168.1.5:8123"
configs:
  "192.168.1.5:8123":
    auth:
      username: jenkins # this is the registry username
      password: welcome1 # this is the registry password
[root@manager ~]#
```

Configurar registry privado – lab Jenkins

2. Iniciar o lab Jenkins e efetuar start do container

- docker start nexus

Primeiro deployment

1. Subir Redis: deployment

```
1  ---
2  apiVersion: apps/v1 # for versions before 1.9.0 use apps/v1beta2
3  kind: Deployment
4  metadata:
5    name: redis-server
6    labels:
7      app: redis
8  spec:
9    selector:
10     matchLabels:
11       app: redis
12     replicas: 1
13     template:
14       metadata:
15         labels:
16           app: redis
17       spec:
18         containers:
19           - name: redis-server
20             image: redis
21             resources:
22               requests:
23                 cpu: 100m
24                 memory: 100Mi
25             ports:
26               - containerPort: 6379
27  ---
```

2. Subir Redis: service

```
28  apiVersion: v1
29  kind: Service
30  metadata:
31    name: redis-server
32    labels:
33      app: redis
34      role: master
35  spec:
36    ports:
37      - port: 6379
38        targetPort: 6379
39    selector:
40      app: redis
```