

ORQUESTRAÇÃO COM KBS



O QUE É KUBERNETES E QUAL SEU PAPEL?

Kubernetes é uma plataforma de orquestração de containers usada para:

- Executar aplicações distribuídas
- Gerenciar deploys
- Reiniciar serviços automaticamente
- Escalar aplicações conforme demanda

Ele funciona como um "sistema operacional para clusters", distribuindo carga entre máquinas e mantendo tudo funcionando mesmo quando um nó cai



DEPLOY, ESCALABILIDADE E GERENCIAMENTO

DEPLOY

- Descrição do app em YAML
- Replicação e atualização automaticamente
- Atualizações feitas sem derrubar o sistema

ESCALABILIDADE

- replicas: N → Sobre N instâncias da aplicação
- Autoscaler pode aumentar/diminuir pods conforme uso de CPU ou RAM

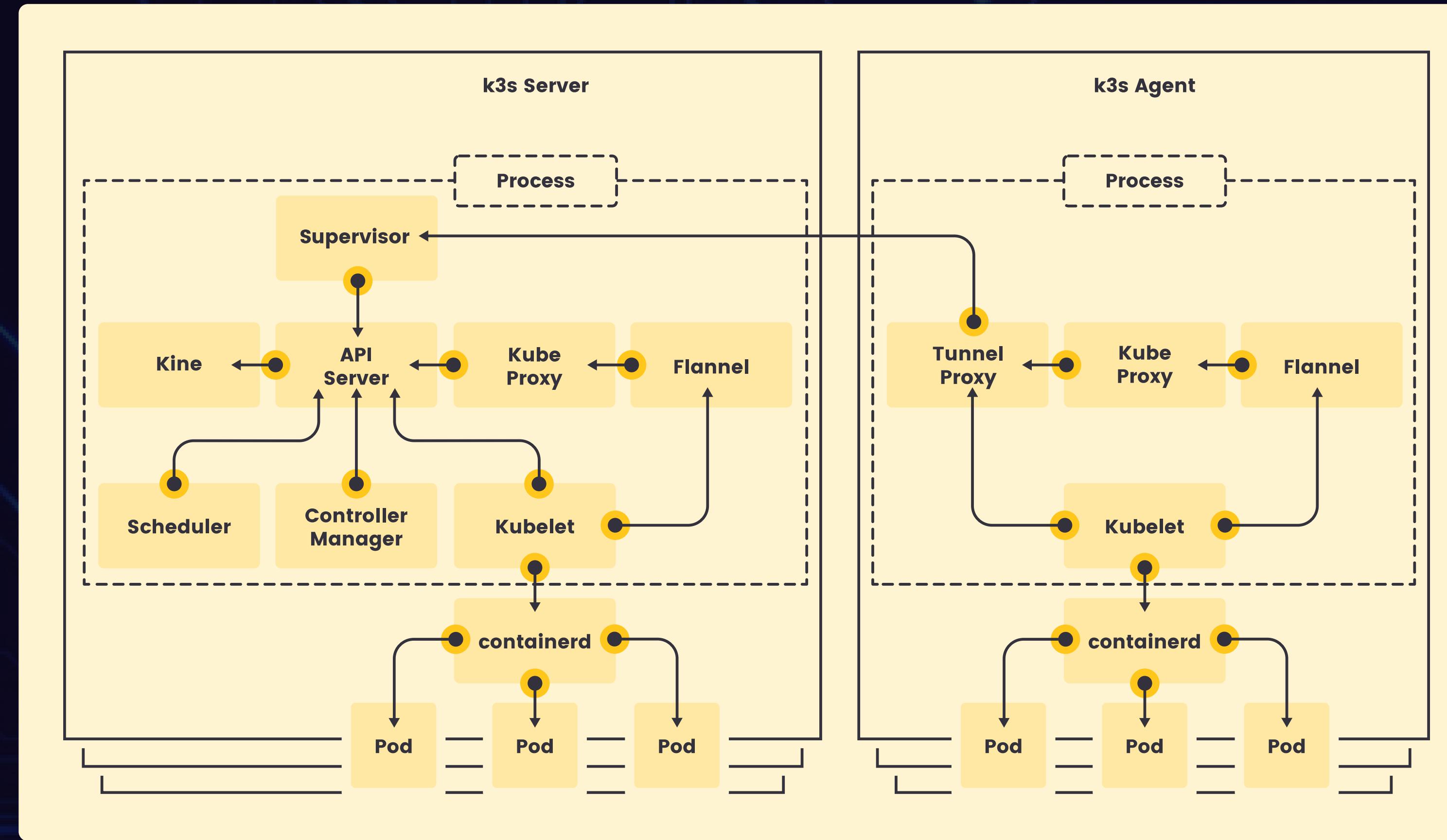
ALTA DISPONIBILIDADE

- Se um pod morrer, é recriado
- Se um node cair, realoca pods automaticamente

GERENCIAMENTO

- Logs centralizados
- Service discovery automático
- Rede interna única entre serviços
- Load balancing nativo

ARQUITETURA K3S





ARQUITETURA K3S - CONCEITOS

POD

- Menor unidade executável
- Contém 1 ou mais containers

SERVICE

- Cria um IP estável para acessar pods
- K3s também possui balanceamento de carga interno

DEPLOYMENT

- Blueprint da aplicação
- Garante número de réplicas
- Atualiza e replica updates automaticamente

NODEPORT

- Expõe um serviço do cluster em uma porta do node (30000-32767)



ARQUITETURA K3S - SERVER

API SERVER

- Porta de entrada do cluster
- Recebe comandos via kubectl e gerencia objetos kubernetes

SCHEDULER

- Decide em qual node cada pod vai rodar
- Mesmo em um único server, gerencia os pods

CONTROLLER MANAGER

- Mantem o estado desejado do cluster
- Garante réplicas
- Reinicia pods
- Atualiza deployments

DATASTORE

- Armazena informações sobre o cluster
- Padrão: SQLite
- High Availability: MySQL, Postgres ou etcd externo



ARQUITETURA K3S - AGENT

KUBELET

- Agente que executa os pods em um node

CONTAINERD

- Substitui o Docker
- Mais leve e otimizado para clusters pequenos

KUBE-PROXY

- Regras de rede e roteamento
- Conecta Services e Pods

FLANNEL + VXLAN

- Rede interna dos pods
- VXLAN sobre UDP 8472
- Cada node recebe um range da rede do cluster



FERRAMENTAS DE MONITORAMENTO

01

K9s

Terminal para navegar no cluster

02

Lens/OpenLens

Dashboard com visualização completa de pods, serviços e logs

03

Prometheus + Grafana

Coleta métricas do cluster e fornece dashboards visuais com CPU, memória e erros

04

Kube-state-metrics

Informações sobre estado de deployments, nodes e pods

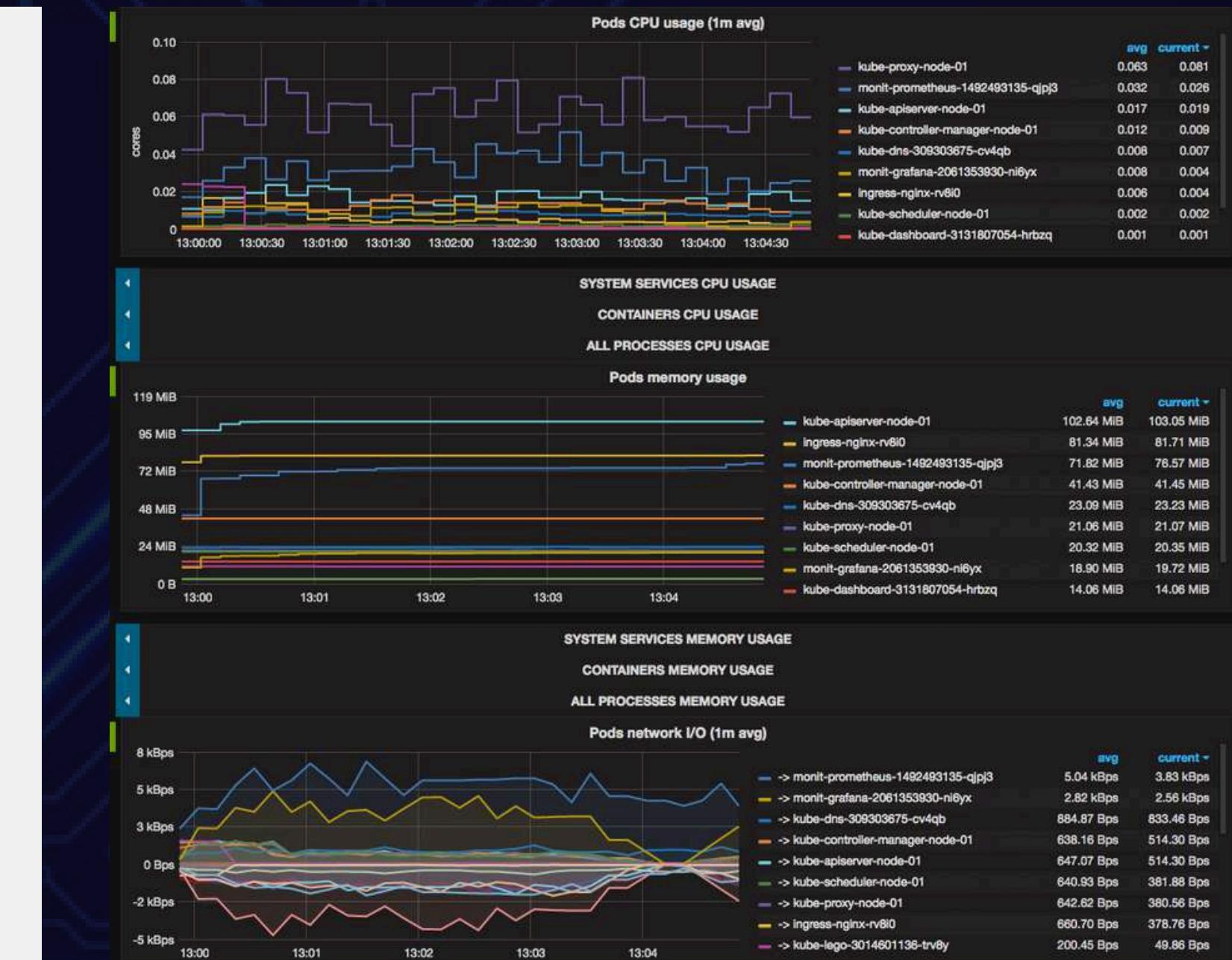
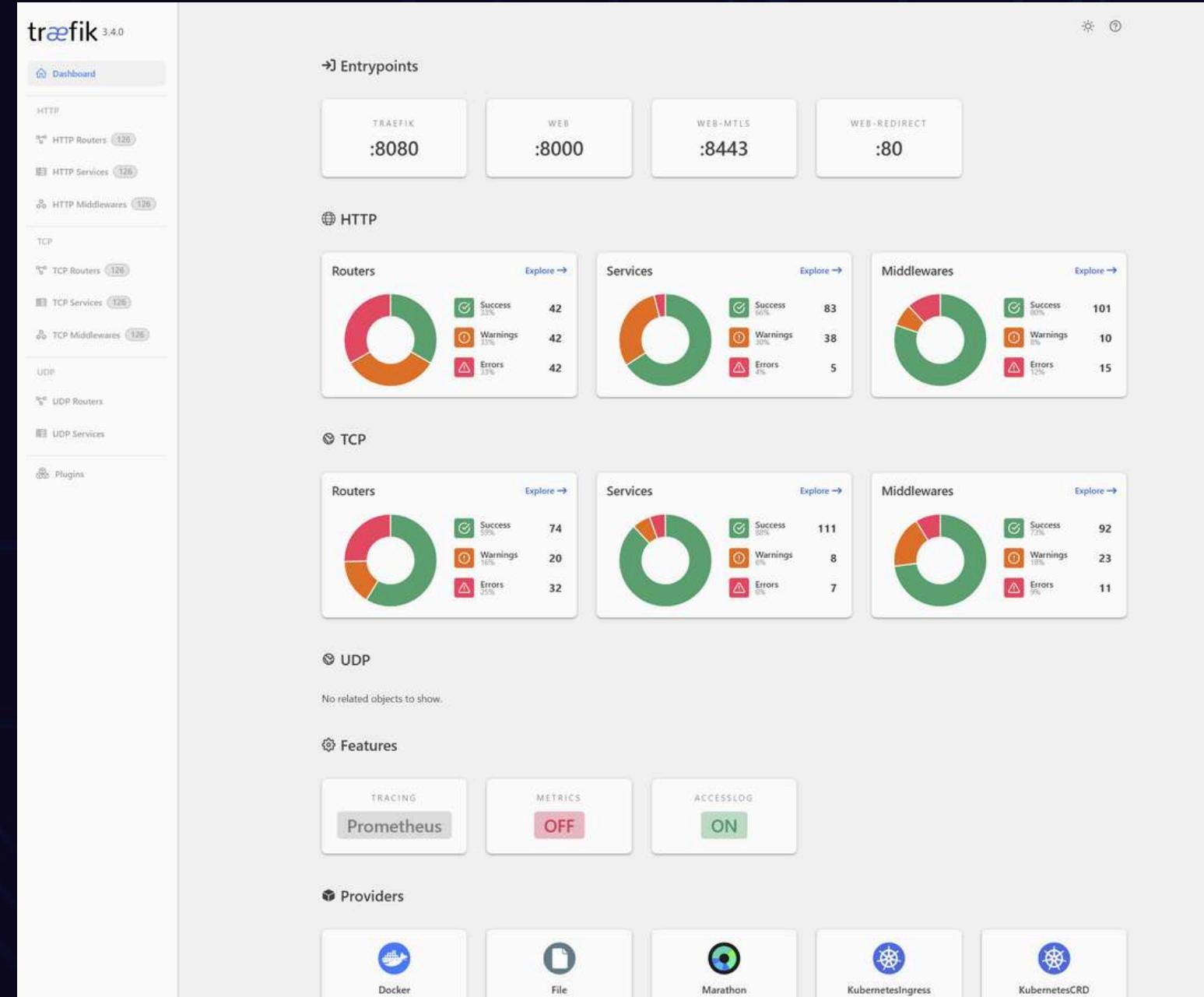
05

Traefik

Ingress controller nativo do k3s, inclui métricas e dashboards



FERRAMENTAS DE MONITORAMENTO



Traefik

Prometheus



FERRAMENTAS DE MONITORAMENTO

The screenshot shows the Lens application interface. On the left is a sidebar with navigation links such as Overview, Pods, Deployments, Daemon Sets, Stateful Sets, Replica Sets, and Replication Controllers. The main area displays four circular dashboards for different resource types: Pods (21), Deployments (1), Daemon Sets (1), and Stateful Sets (1). Below these are smaller circles for Services (1) and Ingresses (1). At the bottom, there's a section for Events with 18 of 230 entries.

Context: minikube <0> all <a> Attach <ctrl-j> Logs (jq)
Cluster: minikube <1> kube-system <ctrl-d> Delete <ctrl-l> Logs <Stern>
User: minikube <2> default <d> Describe <shift-l> Logs Previous
K9s Rev: dev <e> Edit <shift-f> Port-Forward
K8s Rev: v1.17.3 <ctrl-k> Kill <s> Shell
CPU: 5% <l> Logs <y> YAML
MEM: 17%

<0> all <a> Attach <ctrl-j> Logs (jq)
<1> kube-system <ctrl-d> Delete <ctrl-l> Logs <Stern>
<2> default <d> Describe <shift-l> Logs Previous
<e> Edit <shift-f> Port-Forward
<ctrl-k> Kill <s> Shell
<l> Logs <y> YAML

NAMESPACE↑	NAME	READY	RESTART	STATUS	CPU	MEM	%CPU/R	%MEM/R	%CPU/L	%MEM/L	IP	NODE	
default	hello-1582785780-lsrtd	0/1	0	Completed	n/a	n/a	n/a	n/a	n/a	172.17.0.12	minikube		
default	hello-1582785840-rq8hb5	0/1	0	Completed	n/a	n/a	n/a	n/a	n/a	172.17.0.12	minikube		
default	hello-1582785900-4zbkf	0/1	0	Completed	n/a	n/a	n/a	n/a	n/a	172.17.0.12	minikube		
default	jaeger-5bbc8c887-cmj7j	1/1	1	Running	0	7	0	3	0	3	172.17.0.11	minikube	
default	nginx	1/1	1	Running	0	4	0	0	0	0	172.17.0.10	minikube	
default	nginx-6fbddc48c-5kv5p	1/1	0	Running	0	2	0	28	0	14	172.17.0.15	minikube	
default	nginx-6fbddc48c-7xn7j	1/1	0	Running	n/a	n/a	n/a	n/a	n/a	172.17.0.7	minikube		
default	nginx-6fbddc48c-bmqqj	1/1	0	Running	n/a	n/a	n/a	n/a	n/a	172.17.0.13	minikube		
default	nginx-6fbddc48c-jf944	1/1	0	Running	n/a	n/a	n/a	n/a	n/a	172.17.0.12	minikube		
default	nginx-6fbddc48c-xwjnb	1/1	0	Running	0	3	0	39	0	19	172.17.0.14	minikube	
kube-system	coredns-6955765f44-2pkvx	1/1	1	Running	3	7	3	10	0	0	4	172.17.0.2	minikube
kube-system	coredns-6955765f44-wr88k	1/1	1	Running	3	7	3	10	0	0	4	172.17.0.3	minikube
etcd-minikube	etcd-minikube	1/1	1	Running	20	29	0	0	0	0	0	192.168.64.15	minikube
kube-system	fluentd-elasticsearch-vnt25	1/1	1	Running	1	51	1	25	0	25	172.17.0.5	minikube	
kube-system	kube-apiserver-minikube	1/1	1	Running	47	227	18	0	0	0	0	192.168.64.15	minikube
kube-system	kube-controller-manager-minikube	1/1	2	Running	20	35	10	0	0	0	0	192.168.64.15	minikube
kube-system	kube-proxy-sqs9s	1/1	1	Running	0	14	0	0	0	0	0	192.168.64.15	minikube
kube-system	kube-scheduler-minikube	1/1	2	Running	4	12	4	0	0	0	0	192.168.64.15	minikube
kube-system	metrics-server-6754dbc9df-t8x2n	1/1	1	Running	0	13	0	0	0	0	0	172.17.0.8	minikube
kube-system	metrics-server-6754dbc9df-tz7kh	1/1	1	Running	0	10	0	0	0	0	0	172.17.0.6	minikube
kube-system	storage-provisioner	1/1	2	Running	0	14	0	0	0	0	0	192.168.64.15	minikube
kubernetes-dashboard	dashboard-metrics-scraper-7b64584c5c-5tjsh	1/1	1	Running	0	5	0	0	0	0	0	172.17.0.4	minikube
kubernetes-dashboard	kubernetes-dashboard-79d9cd965-wbzvv	1/1	1	Running	0	11	0	0	0	0	0	172.17.0.9	minikube

Lens

K9s



K8S VS K3S



K8s

- Completo, pesado, mais componentes
- etcd como datastore
- exige 2-3 control planes para alta disponibilidade
- Maior custo de RAM e CPU
- Foco em datacenters e nuvens grandes



K3s

- Kubernetes mais leve
- SQLite como datastore
- Binário único
- Ideal para clusters pequenos
- Alta compatibilidade com kubernetes
- Menor consumo de RAM e CPU



VALEU!!

Referências:

- youtube.com
- Rancher
- k3s.io