



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br

Trabalhando com ClusterIP no Kubernetes





Trabalhando com ClusterIP no Kubernetes

Uma aplicação “full-stack” geralmente tem diferentes pods servindo diferentes partes da aplicação.

front-end

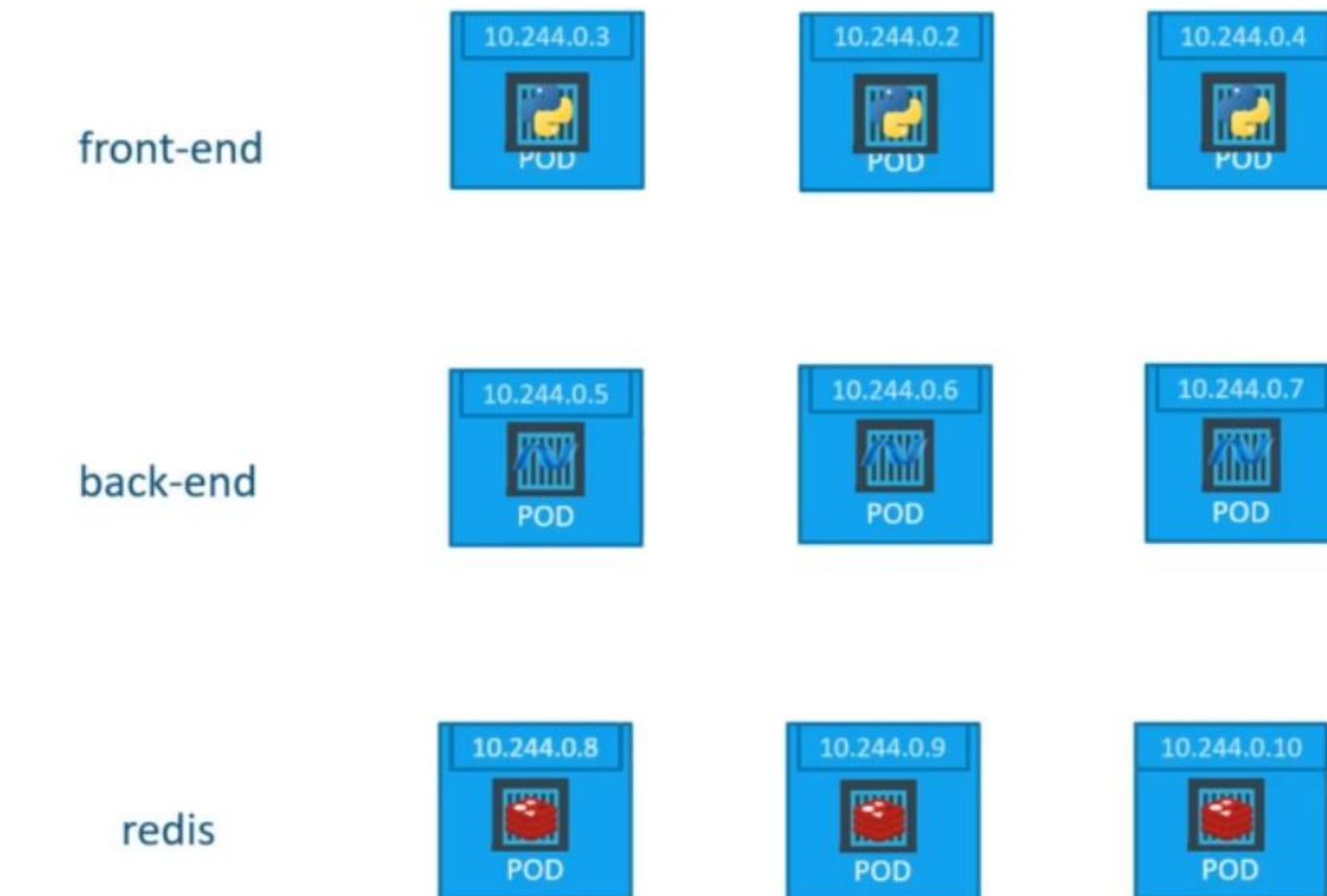
back-end

redis



Trabalhando com ClusterIP no Kubernetes

Uma aplicação “full-stack” geralmente tem diferentes pods servindo diferentes partes da aplicação.

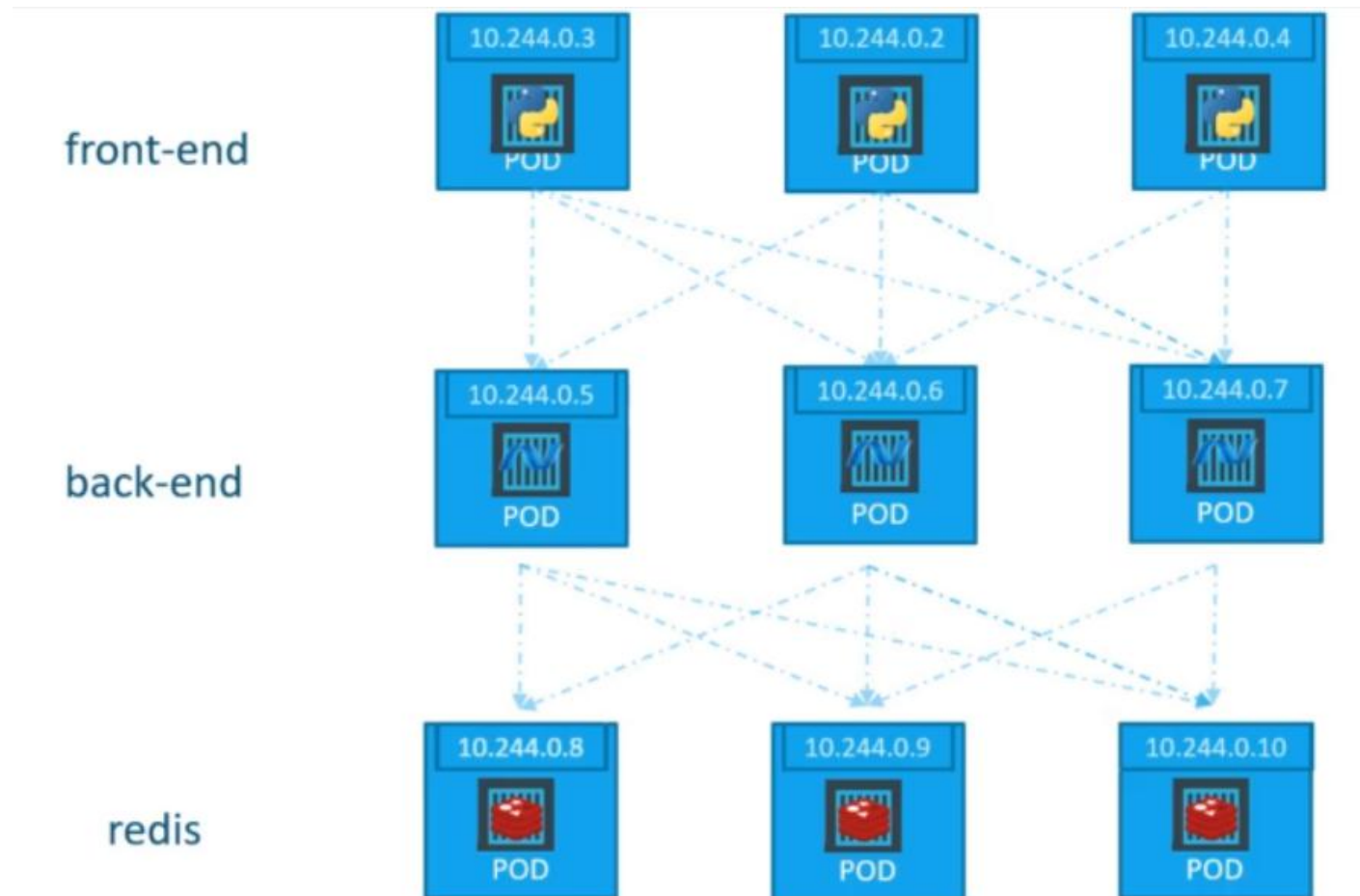




Trabalhando com ClusterIP no Kubernetes

Estes pods precisam se comunicar entre si e esta comunicação não deve sair dos domínios do cluster.

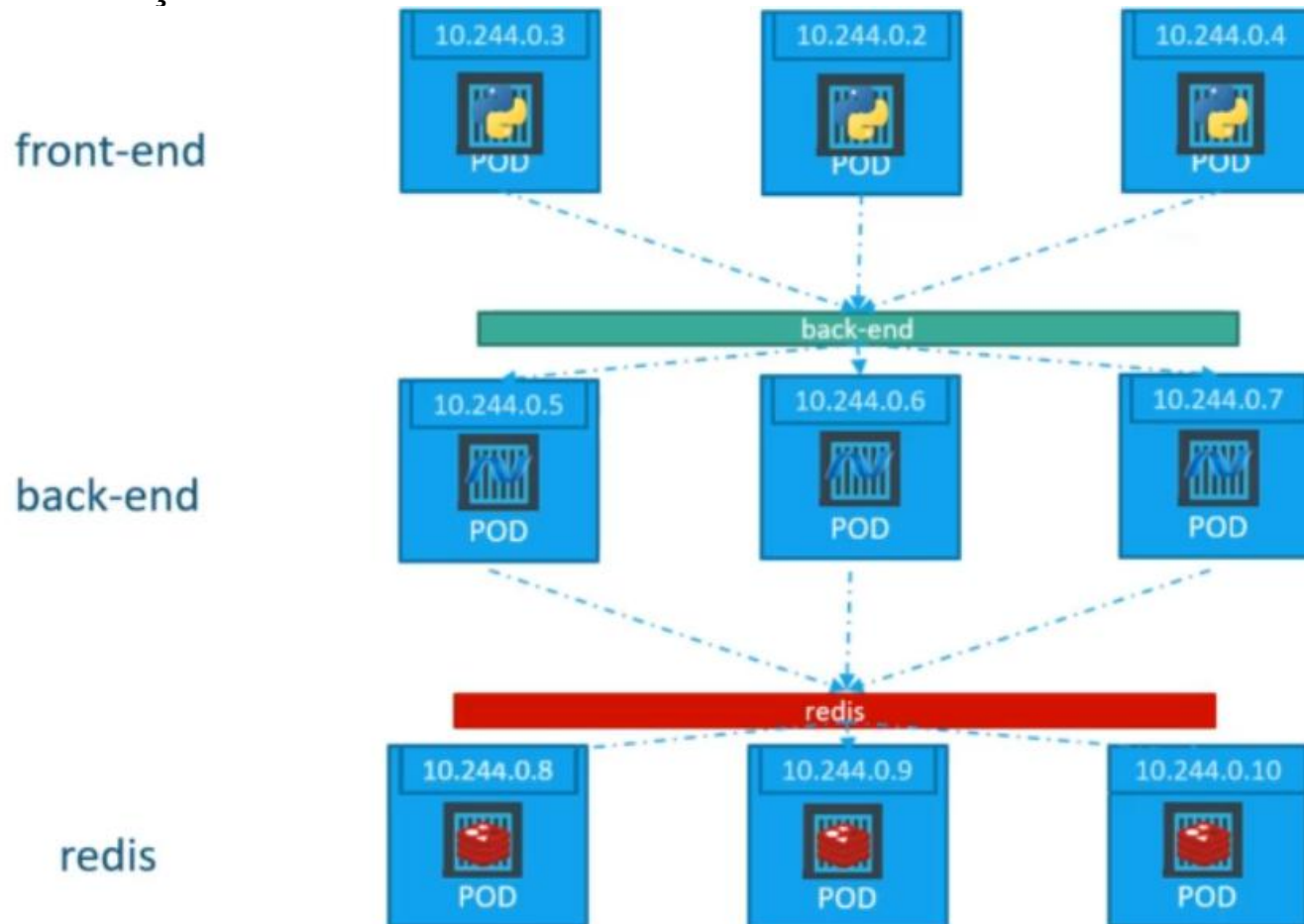
Além disso os pods não possuem IP estático, ou seja, se um pod apresentar problemas e tiver que ser substituído poderá ter um novo endereço IP já que este é dinâmico.





Trabalhando com ClusterIP no Kubernetes

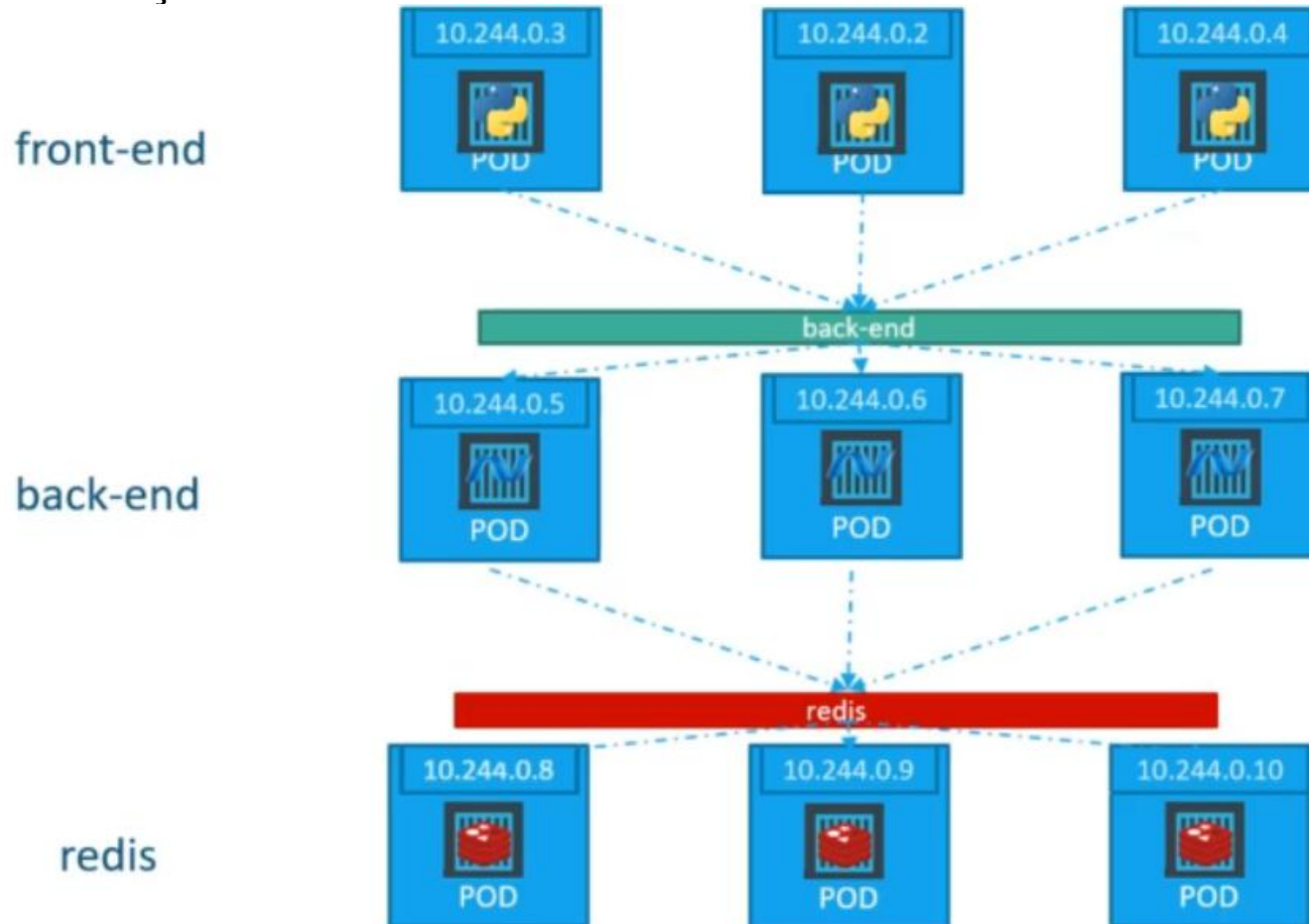
O Service no Kubernetes nos ajuda a “agrupar” os pods que oferecem o mesmo serviço e fornecer uma interface única de acesso aos serviços.





Trabalhando com ClusterIP no Kubernetes

O Service no Kubernetes nos ajuda a “agrupar” os pods que oferecem o mesmo serviço e fornecer uma interface única de acesso aos serviços.



O Service ClusterIP serve justamente para este fim, deixando a comunicação entre os pods funcionarem apenas internamente ao cluster, sem permitir o acesso externo.

Desta forma, podemos criar “microserviços” que juntos, farão parte de toda aplicação.


Cada “microserviço” recebe um nome e um endereço IP na qual deve ser usado pelos outros “microserviços” para realizar o acesso.



Trabalhando com ClusterIP no Kubernetes

Arquivo de definição do Service ClusterIP

```
apiVersion: v1
kind: Service
metadata:
  name: frontend
spec:
  selector:
    type: frontend
  ports:
    - name: frontend
      port: 80
      targetPort: 80
  type: NodePort
```



```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend
  labels:
    app: frontend
    type: frontend
spec:
  template:
    metadata:
      name: frontend
      labels:
        app: frontend
        type: frontend
    spec:
      containers:
        - name: frontend
          image: nginx
      selector:
        matchLabels:
          type: frontend
  replicas: 2
```

O frontend é a parte da aplicação que será acessada pelos clientes/usuários e desta forma precisa ter acesso externo.

Desta forma não podemos usar o ClusterIP aqui e sim o NodePort.



Trabalhando com ClusterIP no Kubernetes

Arquivo de definição do Service ClusterIP

```
apiVersion: v1
kind: Service
metadata:
  name: backend
spec:
  selector:
    type: backend
  ports:
    - name: nginx
      port: 80
  type: ClusterIP
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend
  labels:
    app: backend
    type: backend
spec:
  template:
    metadata:
      name: backend
      labels:
        app: backend
        type: backend
    spec:
      containers:
        - name: nginx
          image: nginx
  selector:
    matchLabels:
      type: backend
  replicas: 2
```



Trabalhando com ClusterIP no Kubernetes

Arquivo de definição do Service ClusterIP

```
apiVersion: v1
kind: Service
metadata:
  name: redis
spec:
  selector:
    type: redis
  ports:
    - name: redis
      port: 6379
  type: ClusterIP
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: redis
  labels:
    app: redis
    type: redis
spec:
  template:
    metadata:
      name: redis
      labels:
        app: redis
        type: redis
    spec:
      containers:
        - name: redis
          image: redis
  selector:
    matchLabels:
      type: redis
  replicas: 1
```



Trabalhando com ClusterIP no Kubernetes

Kubectl

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
geek@university:~/Downloads/secao06$ kubectl create -f deployments/frontend.yaml --save-config --record  
deployment.apps/frontend created  
geek@university:~/Downloads/secao06$ kubectl create -f deployments/backend.yaml --save-config --record  
deployment.apps/backend created  
geek@university:~/Downloads/secao06$ kubectl create -f deployments/redis.yaml --save-config --record  
deployment.apps/redis created  
geek@university:~/Downloads/secao06$
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
geek@university:~/Downloads/secao06$ kubectl create -f services/frontend.yaml --save-config --record  
service/frontend created  
geek@university:~/Downloads/secao06$ kubectl create -f services/backend.yaml --save-config --record  
service/backend created  
geek@university:~/Downloads/secao06$ kubectl create -f services/redis.yaml --save-config --record  
service/redis created  
geek@university:~/Downloads/secao06$
```




Trabalhando com ClusterIP no Kubernetes

Kubectl

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

geek@university:~/Downloads/secao06$ kubectl get all
NAME                                READY    STATUS    RESTARTS    AGE
pod/backend-956cf8654-89z5x         1/1      Running   0            5m39s
pod/backend-956cf8654-jfzld         1/1      Running   0            5m39s
pod/frontend-75596f9fc4-7x5z4       1/1      Running   0            5m45s
pod/frontend-75596f9fc4-z7p9m       1/1      Running   0            5m45s
pod/redis-749f4cd47-5r2k8           1/1      Running   0            5m32s

NAME                                TYPE          CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
service/backend                     ClusterIP      10.99.10.69    <none>          80/TCP           5m21s
service/frontend                     NodePort       10.100.46.124  <none>          80:32417/TCP     5m26s
service/kubernetes                   ClusterIP      10.96.0.1      <none>          443/TCP          13m
service/redis                       ClusterIP      10.100.10.185  <none>          6379/TCP         5m15s

NAME                                READY    UP-TO-DATE    AVAILABLE    AGE
deployment.apps/backend              2/2      2              2            5m39s
deployment.apps/frontend              2/2      2              2            5m45s
deployment.apps/redis                 1/1      1              1            5m32s

NAME                                DESIRED    CURRENT    READY    AGE
replicaset.apps/backend-956cf8654    2          2          2        5m39s
replicaset.apps/frontend-75596f9fc4  2          2          2        5m45s
replicaset.apps/redis-749f4cd47      1          1          1        5m32s

geek@university:~/Downloads/secao06$
```



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br