

# 通过端口转发访问集群中的应用程序

本页向您展示如何使用 `kubectl port-forward` 命令连接到运行在 Kubernetes 集群中的 Redis 服务器。这种类型的连接对于数据库调试很有帮助。

## 创建一个 Pod 来运行 Redis 服务器

```
[root@vlnx251101 redis]# vim redis-master.yaml
```

```
apiVersion: v1
kind: Pod
metadata:
  labels:
    name: redis
    redis-sentinel: "true"
    role: master
  name: redis-master
spec:
  containers:
    - name: master
      image: k8s.gcr.io/redis:v1
      env:
        - name: MASTER
          value: "true"
      ports:
        - containerPort: 6379
```

```

resources:
  limits:
    cpu: "0.1"
  volumeMounts:
    - mountPath: /redis-master-data
      name: data
- name: sentinel
  image: kubernetes/redis:v1
  env:
    - name: SENTINEL
      value: "true"
  ports:
    - containerPort: 26379
volumes:
- name: data
  emptyDir: {}

```

```

[root@vlnx251101 redis]# kubectl create -f redis-
master.yaml
pod/redis-master created

```

```

[root@vlnx251101 redis]# kubectl get pods

```

NAME	READY	STATUS	RESTARTS	AGE
redis-master	2/2	Running	0	1m

```
[root@vlnx251101 redis]# kubectl get pods redis-master --  
template='{{(index (index .spec.containers 0).ports  
0).containerPort}}'{{"\n"}}'  
6379
```

## 将本地端口转发到 Pod 中的端口

```
[root@vlnx251101 redis]# yum install redis socat
```

```
[root@vlnx251101 redis]# kubectl port-forward redis-master  
6379:6379  
Forwarding from 127.0.0.1:6379 -> 6379  
Forwarding from [::1]:6379 -> 6379
```

```
[root@vlnx251101 ~]# redis-cli  
127.0.0.1:6379> ping  
PONG
```

创建连接，将本地的 6379 端口转发到运行在 Pod 中的 Redis 服务器的 6379 端口。有了这个连接您就可以在本地工作站中调试运行在 Pod 中的数据库。