Init-Containers

» Init-Containers

Overview

Create a manifest file for a Pod that contains one application container and two init containers.

```
cat > init-containers.yaml <<EOF
apiVersion: v1
kind: Pod
metadata:
  name: myapp
  labels:
    app: myapp
spec:
  containers:
  - name: myapp-container
    image: busybox
    command: ['sh', '-c', 'echo The app is running! && sleep 3600']
  initContainers:
  - name: init-myservice
    image: busybox
    command: ['sh', '-c', 'until nslookup myservice; do echo waiting for myservice; sleep 2; done;']
  - name: init-mydb
    image: busybox
    command: ['sh', '-c', 'until nslookup mydb; do echo waiting for mydb; sleep 2; done;']
EOF
```

Create the Pod

```
kubectl create -f init-containers.yaml
```

Verify the Pod is not ready.

```
kubectl get pods -l app=myapp
```

Observe the logs within the first init container. `nslookup` will continue to run in this container until it can resolve `myservice` .

```
kubectl logs -f myapp -c init-myservice
```

Create a manifest file for the `myservice` Service.

```
cat > init-myservice.yaml <<EOF
kind: Service
apiVersion: v1
metadata:
  name: myservice
  labels:
    app: myapp
spec:
  ports:
    - protocol: TCP
      port: 80
      targetPort: 9376
EOF
```

Create the service.

```
kubectl create -f init-myservice.yaml
```

Observe the first init-container's logs again.

```
kubectl logs -f myapp -c init-myservice
```

Now that it can resolve the name, the second init-container has kicked off.

```
kubectl logs -f myapp -c init-mydb
```

Create a manifest file for mydb service.

```
cat > init-mydb.yaml <<EOF
kind: Service
apiVersion: v1
metadata:
  name: mydb
  labels:
    app: myapp
spec:
  ports:
    - protocol: TCP
      port: 80
      targetPort: 9377
EOF
```

Create the mydb service.

```
kubectl create -f init-mydb.yaml
```

Observe the second container's logs again.

```
kubectl logs -f myapp -c init-mydb
```

As soon as the name resolves, verify the primary container is running.

```
kubectl logs -f myapp -c myapp-container
```

Clean Up

Delete the pod and services that we defined for the application.

```
kubectl delete pod,svc -l app=myapp
```

View any remaining resources.

```
kubectl get all
```