Deployments

» Deploying a Simple Application

Let’s Create an Example Application

Create the deployment

We will deploy a simple, stateless Hello World application written in **golang**.

Create the deployment object manifest file.

```
cat > simple-deployment.yaml<<EOF
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: simple-deployment
  namespace: myproject
  labels:
    k8s-app: simple
spec:
  replicas: 3
  template:
    metadata:
      labels:
        k8s-app: simple
    spec:
      serviceAccountName: useroot
      containers:
        - name: hieveryone
```

```
          image: quay.io/coreostrainme/hieveryone:1.0.0
          imagePullPolicy: Always
          ports:
            - name: helloworldport
              containerPort: 80
EOF
```

Create the deployment based off our newly created deployment object manifest file.

```
oc create -f simple-deployment.yaml --record=true
```

Deployments keep track of any changes we make to our pod template and allow us to pause, resume, and rollback to previous configurations.

View the rollout help.

```
oc rollout --help
```

View rollout history.

```
oc rollout history deployment simple-deployment
```

Get details about revision 1

```
oc rollout history deployment simple-deployment --revision=1
```

Verify the deployment was successfully created.

```
oc get deployments
```

View the pods associated with the deployment.

```
oc get pods
```

Create the service

We will now expose this deployment as a Service.

Create the service object manifest file.

```
cat > simple-service.yaml<<EOF
kind: Service
apiVersion: v1
metadata:
  name: simple-service
  namespace: myproject
spec:
  selector:
    k8s-app: simple
  ports:
  - protocol: TCP
    port: 8080
    targetPort: 80
EOF
```

Create the deployment based off our newly created deployment object manifest file.

```
oc create -f simple-service.yaml
```

Verify the service was successfully created.

```
oc get service
```

Expose the Service by creating the OpenShift Route manifest.

```
cat > simple-service-route.yaml <<EOF
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: simple-service-route
  namespace: myproject
spec:
  port:
    targetPort: 80
  to:
    kind: Service
    name: simple-service
    weight: 100
  wildcardPolicy: None
EOF
```

Create the route.

```
oc create -f simple-service-route.yaml
```

Fetch the URL for the Openshift Route.

```
ROUTE_HOST_URL=`oc get route simple-service-route -o jsonpath='{.spec.host}'`
echo $ROUTE_HOST_URL
```

Verify web app connectivity.

```
curl $ROUTE_HOST_URL
```