

Ingress Controller

Ingress + IngressController (especially required in hybrid K8s + OCP environments).

Project Creation

```
oc new-project ingress-lab
```

New App Deployment

```
oc new-app --name=hello \  
  quay.io/openshift-examples/hello-front
```

Verify:

```
oc get pods  
oc expose deploy/hello --port=8080  
oc get svc
```

Custom Wide Ingress Controller:

ingresscontroller-custom.yaml

```
apiVersion: operator.openshift.io/v1  
kind: IngressController  
metadata:  
  name: custom  
  namespace: openshift-ingress-operator  
spec:  
  domain: apps.custom.example.com  
  replicas: 2  
  endpointPublishingStrategy:  
    type: LoadBalancerService
```

Apply:

```
oc apply -f ingresscontroller-custom.yaml
```

Check status:

```
oc -n openshift-ingress get pods -l ingresscontroller=custom
```

Get LoadBalancer:

```
oc -n openshift-ingress get svc
```

```
student@workstation ~]$ oc -n openshift-ingress get svc  
NAME                                TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)  
router-internal-custom             ClusterIP      172.30.89.29     <none>            80/TCP,443/TCP,1936/TCP  
router-internal-default             ClusterIP      172.30.6.208     <none>            80/TCP,443/TCP,1936/TCP
```

Ingress Resource Creation

This is pure Kubernetes-style.

File: ingress-hello.yaml

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: hello-ingress
  annotations:
    kubernetes.io/ingress.class: openshift-custom
spec:
  rules:
    - host: hello.apps.custom.example.com
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: hello
                port:
                  number: 8080
```

Apply:

```
oc apply -f ingress-hello.yaml
```

```
curl -k http://hello.apps.custom.example.com
```

TLS configuration for Ingress

```
openssl req -x509 -newkey rsa:2048 -nodes \
  -keyout tls.key -out tls.crt \
  -subj "/CN=hello.apps.custom.example.com" -days 365
```

Secret

```
oc create secret tls hello-tls \
  --cert=tls.crt --key=tls.key -n ingress-lab
```

Add TLS to Ingress

Update the file:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: hello-ingress
  annotations:
    kubernetes.io/ingress.class: openshift-custom
spec:
  tls:
    - hosts:
      - hello.apps.custom.example.com
      secretName: hello-tls
  rules:
    - host: hello.apps.custom.example.com
      http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: hello
                port:
                  number: 8080
```

Apply:

```
oc apply -f ingress-hello.yaml
```

Test:

```
curl -k https://hello.apps.custom.example.com
```

Routing Exposure

Add another app:

```
oc new-app --name=api \
  quay.io/openshift-examples/hello-api
oc expose deploy/api --port=8080
```

Create Ingress:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: multi-path-ingress
  annotations:
    kubernetes.io/ingress.class: openshift-custom
spec:
  rules:
    - host: app.apps.custom.example.com
      http:
        paths:
          - path: /web
            pathType: Prefix
            backend:
```

```
    service:
      name: hello
      port:
        number: 8080
  - path: /api
    pathType: Prefix
    backend:
      service:
        name: api
        port:
          number: 8080
```

Apply:

```
oc apply -f multi-path-ingress.yaml
```

Test:

```
curl http://app.apps.custom.example.com/web
curl http://app.apps.custom.example.com/api
```

```
[student@workstation ~]$ vim multi-path-ingress.yaml
[student@workstation ~]$ vim multi-path-ingress.yaml
[student@workstation ~]$ oc apply -f multi-path-ingress.yaml
Warning: annotation "kubernetes.io/ingress.class" is deprecated, please use 'spec.ingressClassName' instead
ingress.networking.k8s.io/multi-path-ingress created
[student@workstation ~]$ curl http://app.apps.custom.example.com/web

curl http://app.apps.custom.example.com/api
curl: (6) Could not resolve host: app.apps.custom.example.com
curl: (6) Could not resolve host: app.apps.custom.example.com
[student@workstation ~]$ oc delete ingresscontroller custom -n openshift-ingress-operator
ingresscontroller.operator.openshift.io "custom" deleted
[student@workstation ~]$ oc -n openshift-ingress get pods
NAME                                READY   STATUS    RESTARTS   AGE
router-default-56cf89f99f-gvtlg    1/1     Running   12 (4h2m ago)  669d
```