

API-CDN On Openshift

Laurent Valeyre

Auguste 2018

Contents

1	Multiple Projects Pipeline	1
2	Build, Tag, Promote	1
2.1	Create Projects	2
2.2	Modify RBAC	2
2.3	Deploy Jenkins and Our Pipeline Definition	3
2.4	Deploy Our Sample Application	3
2.5	Run Our Pipeline Deployment	4

1 Multiple Projects Pipeline

2 Build, Tag, Promote

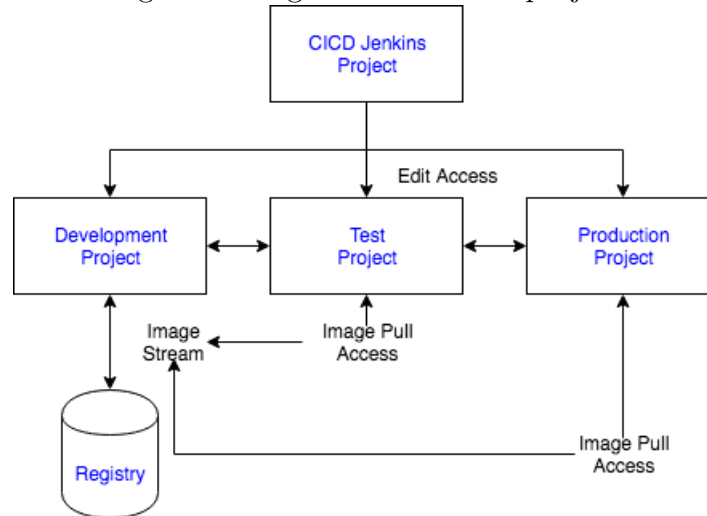
CICD Containing our Jenkins instance.

Development For building and developing our application images.

Testing For testing our application

Production Hosting our production application

Figure 1: diagram of different projects



2.1 Create Projects

```
$ oc login -u developer -p developer
$ oc new-project cicd --display-name='CICD Jenkins' \
--description='CICD Jenkins'
$ oc new-project development --display-name='Development' \
--description='Development'
$ oc new-project testing --display-name='Testing' \
--description='Testing'
$ oc new-project production --display-name='Production' \
--description='Production'
```

2.2 Modify RBAC

Let's add in RBAC to our projects to allow the different service accounts to build, promote, and tag images. First we will allow the cicd project's Jenkins service account edit access to all of our projects:

```
$ oc policy add-role-to-user edit system:serviceaccount:cicd:jenkins \
-n development
$ oc policy add-role-to-user edit system:serviceaccount:cicd:jenkins \
-n testing
```

```
$ oc policy add-role-to-user edit system:serviceaccount:cicd:jenkins \
-n production
```

Now we want to allow our testing and production service accounts the ability to pull images from the development project:

```
$ oc policy add-role-to-group system:image-puller system:serviceaccounts:testing
-n development
$ oc policy add-role-to-group system:image-puller system:serviceaccounts:production
-n development
```

2.3 Deploy Jenkins and Our Pipeline Definition

We deploy a Jenkins ephemeral instance to our *cicd* project.

```
$ oc project cicd
$ oc new-app --template=openshift/jenkins-persistent
$ oc status
```

Let's create the pipeline itself.

```
$ oc create -n cicd -f \
https://raw.githubusercontent.com/devops-with-openshift/pipeline-configs/master/
```

2.4 Deploy Our Sample Application

```
$ oc project development
$ oc create new-app --name=myapp \
openshift/php:5.6~https://github.com/devops-with-openshift/cotd.git#master
$ oc expose svc/myapp
```

We'll look for the Docker Registry IP address.

```
$ oc get imagestream -n development
```

NAME	DOCKER REPO	TAGS	UPDATED
myapp	172.30.1.1:5000/development/myapp		

We'll create the deployment config in the *testing* project.

```
$ oc project testing
$ oc create deploymentconfig \
--image= 172.30.1.1:5000/development/myapp:promoteQA myapp
$ oc rollout cancel dc/myapp
```

And to be certain that it will pull the image

```
$ oc patch dc/myapp \
-p '{"spec":{"template":{"spec":{"containers":[{"name":"default-
container","imagePullPolicy":"Always"}]}}}}'
$ oc rollout cancel dc/myapp
```

Finally we expose the service and route

```
$ oc expose dc myapp --port=8080
$ oc expose svc/myapp
```

We'll do the same process to the *production* project.

```
$ oc project production
$ oc create deploymentconfig \
--image= 172.30.1.1:5000/development/myapp:promotePRD myapp
$ oc rollout cancel dc/myapp
$ oc patch dc/myapp \
-p '{"spec":{"template":{"spec":{"containers":[{"name":"default-
container","imagePullPolicy":"Always"}]}}}}'
$ oc rollout cancel dc/myapp
$ oc expose dc myapp --port=8080
$ oc expose svc/myapp
```

We are using two separate image tags: *promoteQA* for testing promotion and *promotePRD* for production promotion.

2.5 Run Our Pipeline Deployment

```
$ oc start-build pipeline -n cicd
```

List of Figures

1	diagram of different projects	1
---	---	---