Standard Operating Procedure

*New Project Creation*

## PURPOSE

The purpose of this standard operating procedure (SOP) is to provide a detailed step-by-step procedure on how to create new projects in OpenShift.

## PROCEDURE

1. Verify the requested project prefix is listed here: <https://confluence.marriott.com/display/RUNTEAM/OSE+Project+Owners>
   1. If not, request a DUD (cost center) from the project requester before proceeding further
   2. Inform Marriott L3, Andrew Shih
   3. Update the above confluence page with the new prefix and the DUD once available and proceed. The project prefix is used for all projects that belong to the same DUD.
2. SSH into the master node 1 of the requested environment
3. Switch to OSE service account, ex: svc-vxby-ose (change based on environment). If working with CR Perf environment, switch to root user

sudo su - svc-vxby-ose

(Skip Steps 4-6 if user already exists)



htpasswd /etc/origin/master/htpasswd <username> on all **3 master nodes** as **root** user**.** For creating the username, follow the naming convention, <project prefix><admin>, (ex: ramadmin, dspadmin). The username would show up after logging in (step 5). Check all the names usingoc get user

1. systemctl list-units | grep atomic

to check the following services are running on all three masters

atomic-openshift-master-api.service

atomic-openshift-master-controllers.service

atomic-openshift-node.service

1. Check new login from CLI (Command Line Interface)

oc login

Enter user name and password at login challenge. Should be able to login

1. Login or confirm login to system:admin user:

oc login --username=system:admin or oc whoami

1. oc new-project <project name>
   1. Project names are set by OSE team. They should follow the naming convention <project prefix>-<environment><number> (ex: ram-prod1, ram-prodengage1, dsp-dev1, dsp-dev2)
2. oc get projects | grep <project name>

to make sure that the project is active

1. oadm policy add-role-to-user <admin|view> <username> -n <project name>
   1. In Production, the value will **always** be view, not admin
2. Login to OpenShift console using the new or existing username
3. In master01 node under /home/svc-vxby-ose/OpenShift/environments/<server name>/artifacts/quota create new folder <project name> (change *vxby* based on location)

mkdir <project name>

1. cd <project name>
2. oc edit ns/<project name>

make sure to have the correct <project name>

To get the region and zones, this command can get used outside of this file:

oc get nodes -o wide --show-labels | grep node

add

**openshift.io/node-selector: region=primary,zone=<zone to which the project needs to be deployed>** (i.e. openshift.io/node-selector: region=primary,zone=<zone>)

to the file.

Update **openshift.io/requester** to requester name with which clusterresourcequota is created.

After updating openshift.io/requester project will fall under particular clusterresourcequota

1. Create local yaml files for Limit Range and Resource Quota by copying default yaml files (still in the /quota/<project name> folder on master01 node). Make sure to follow the naming convention

cp <existing\_LimitRange>.yaml <new\_LimitRange>.yaml

cp <existing\_ResourceQuota>.yaml <new\_ResourceQuota>.yaml

For Limit Range:

oc create -f <project name-limits>.yaml

make sure to have project name correctly in the yaml file and apply the appropriate values in the file.

* 1. oc get limits

to verify *that* the limit range is created

* 1. oc describe limits <limit range name>

to check the details of the limit range

* 1. Log into OpenShift console to double check the limit range

(Skip Step 16 if quota already exists, as this is tied to the user not project)

1. For Resource Quota:
   1. oc create -f <project name\_ResourceQuota.yaml>
   2. oc describe clusterresourcequota

to verify the details of cluster resource quota. Go onto the console to double check

1. oc get sa

to check the service accounts

(Skip Steps 18-19 if service account already exists)

1. Create Jenkins service account:

oc create sa <project prefix>jenkinssa

1. oc get sa

to verify

1. oc policy add-role-to-group edit system:serviceaccounts:<project containing the service account> -n <project being created> (i.e.: if ram-dev1 is the newly created project, its service account needs to get access to its own project. oc policy add-role-to-group edit system:serviceaccounts:ram-dev1 -n ram-dev1)
2. oc sa get-token <sa name>

copy the token

1. oc login --token=<copied token>
2. oc login --username=system:admin
3. Go to /home/svc-vxby-ose/OpenShift/environments/<servername>/artifacts/nfsmounts (change *vxby* based on location
4. For PV/PVC:
   1. Request NFS export from Toan. Assign Group: Idea-Infrastructure-Automation-Cloud-Support. It can be a mount (i.e. /u01) or a path. If server IP is given, compare that with the yaml file from the next step
   2. Create local yaml files for PV/PVC by copying default yaml file (in the /nfsmounts folder on master01 node). Make sure to follow the naming convention

cp -fr <existing project name> <new project name>

* 1. cd <new project name>
  2. cp <existing\_pv>.yaml <new\_pv>.yaml
  3. edit the <new\_pv>.yaml

make sure storages are the same and metadata names and spec volumeName are all the same up until -pv (as for some project, if typed in “pvc”, pv/pvc would not bound)

* 1. oc create -f <project name-path-pv.yaml>
  2. oc get pvc

to verify that pvc is created

1. To verify PV/PVC:
   1. Deploy Jenkins app:

oc new-app --docker-image=jenkins --name=jenkins

Output would look like this:

--> Found Docker image cd14cec (10 weeks old) from Docker Hub for "jenkins"

\* An image stream will be created as "jenkins:latest" that will track this image

\* This image will be deployed in deployment config "jenkins"

\* Ports 50000/tcp, 8080/tcp will be load balanced by service "jenkins"

\* Other containers can access this service through the hostname "jenkins"

\* This image declares volumes and will default to use non-persistent, host-local storage.

You can add persistent volumes later by running 'volume dc/jenkins --add ...'

--> Creating resources with label app=jenkins ...

imagestream "jenkins" created

deploymentconfig "jenkins" created

service "jenkins" created

--> Success

Run 'oc status' to view your app.

* 1. Change dc to attach pvc:

oc volume dc/jenkins --add --name=v1 -t pvc --claim-name=kim-dev1-u134-pvc --mount-path=/data --overwrite

This should return deploymentconfigs/Jenkins

* 1. Go inside the cotainer and check /data partition

oc rsh <jenkins name>

df –h

Possible output is:

Filesystem Size Used Avail Use% Mounted on

/dev/mapper/docker-253:0-33576106-6bd76938d81fd14636d64e3dcb3179b42d24a5ec4097e1939eb88b90a6d381da 10G 755M 9.3G 8% /

tmpfs 16G 0 16G 0% /dev

tmpfs 16G 0 16G 0% /sys/fs/cgroup

10.224.211.95:/u134 2.0G 6.0M 1.8G 1% /data

## VALIDATION

Follow the validation steps in “Procedure”.

**Document Version Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Effective Date | Reviewed by | Approved by |
| 1.0 | 10/1/2018 | KyungIn Kim | Yashi Kumar |
|  |  |  |  |

Document Modification History & Revision Log

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Version | Date Modified | Revised by | Role | Section Affected | Reference | Remarks |
| 1.0 |  |  |  |  |  |  |
| 1.1 |  |  |  |  |  |  |