

Advanced Deployment with OpenShift - Homework

This is a document to provide the necessary information about running the script which is used to deploy the OpenShift cluster and complete the CICD pipeline which is needed to successfully complete the homework assignment.

Git repository with the homework scripts:

https://github.com/thednd1992/ocp_advanced_deployment_homework.git

Step	Description	Command
1	Login to VM	<code>ssh -i ~/.ssh/id_rsa djordje.gajisin-devoteam.com@bastion.6a28.example.opentlc.com</code>
2	Switch to root user	<code>sudo -i</code>
3	Clone the Git repository	<code>git clone https://github.com/thednd1992/ocp_advanced_deployment_homework.git</code>
4	Run the ansible playbook	<code>ansible-playbook ./ocp_advanced_deployment_homework/homework.yaml</code>
5	Uninstall the cluster	<code>sh ./ocp_advanced_deployment_homework/scripts/uninstall.sh</code>

The homework.yaml script automatically deploys the OpenShift cluster, creates PVs with different sizes (5G and 10G) and creates the different users requested in the assignment. The script also deploys the NodeJS-Mongo-Persistent app as a smoke test to see the ability to deploy a simple app. The CICD pipeline is created in the task-dev project and it is promoted to the task-prod project automatically through the pipeline. In the end the scripts provide two groups with the requested users and creates the limit ranges.

The following table represents the projects, their routes and login credentials.

Service name	Route	Login credentials
gogs	gogs-tasks-dev.apps.6a28.example.opentlc.com	gogs/gogs
jenkins	jenkins-tasks-dev.apps.6a28.example.opentlc.com	andrew/r3dh4t1!
nexus	nexus-tasks-dev.apps.6a28.example.opentlc.com	admin/admin123
sonarqube	sonarqube-tasks-dev.apps.6a28.example.opentlc.com	admin/admin
tasks	tasks-tasks-prod.apps.6a28.example.opentlc.com	
node-js-app	nodejs-mongo-persistent-smoke-test.apps.6a28.example.opentlc.com	

Release version: 3.11.16

Instructor: Jindrich Kana

Venue: Garni Hotel Centar, Novi Sad, Serbia

Participant: Djordje Gajisin (djordje.gajisin@devoteam.com)

Gogs

gogs-tasks-dev.apps.6a28.example.opentlc.com/gogs



Dashboard

Issues

Pull Requests

Explore



gogs

✉ admin@gogs.com

🕒 Joined on Dec 04, 2018

👤 0 Followers - 0 Following

📁 Repositories

🔗 Public Activity

openshift-tasks


Updated 23 minutes ago

★ 0 🍴 0


Jenkins


← → ↻


Not secure | https://jenkins-tasks-dev.apps.6a28.example.opentlc.com


 Jenkins


Jenkins ▶


 New Item


 People


 Build History


 Manage Jenkins

 My Views

 Open Blue Ocean

 Lockable Resources

 Credentials

 New View

Build Queue

No builds in the queue.

Build Executor Status

1 Idle





2 Idle

3 Idle

4 Idle

5 Idle

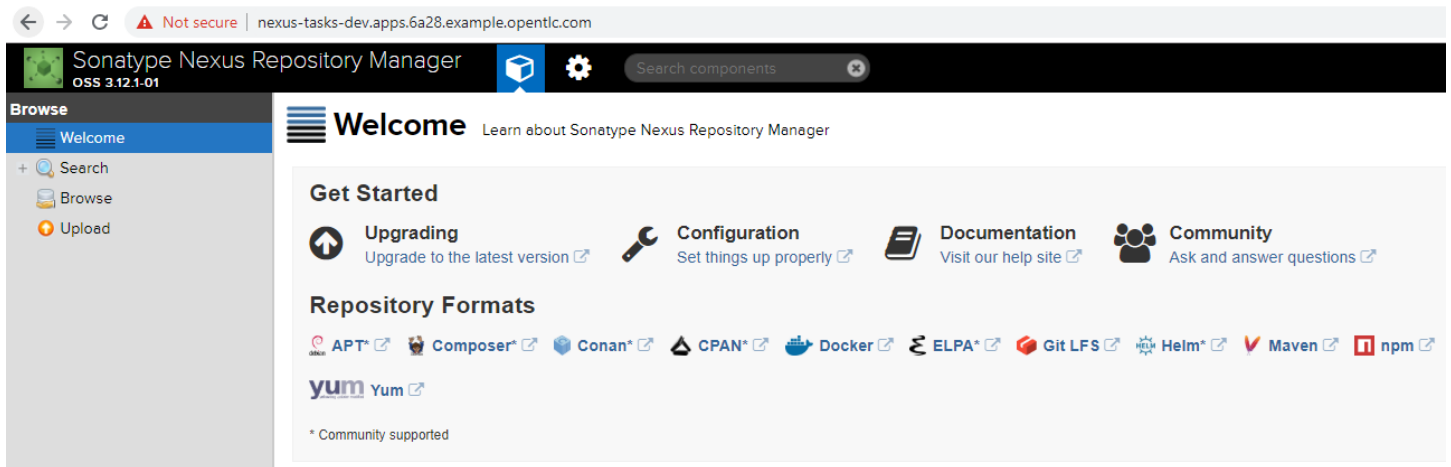
All +

S	W	Name ↓	Last Success
		OpenShift Sample	N/A
		tasks-dev	N/A

Icon: [S](#) [M](#) [L](#)

4

Nexus



SonarQube

sonarqube

ProjectsIssuesRulesQuality ProfilesQuality GatesAdministration

sonarqube-tasks-dev.apps.6a28.example.opentlc.com/projects

My FavoritesAll

Filters

Quality Gate

Passed

Warning

Failed

1

0

0

Reliability (Bugs)

A

B and worse

C and worse

D and worse

E

0

1

1

0

0

Security (Vulnerabilities)

A

B and worse

C and worse

D and worse

E

0

1

0

0

0

Perspective: Overall Status

Sort by: Name

Search by project

JBoss EAP - Tasks JAX-RS App

Passed

2

C

Bugs

1

B

Vulnerabilities

17

A

Code Smells

5.9%

Coverage

0.0%

Duplications

1 of 1 shown

Task-prod

← → ↻

Not secure | tasks-tasks-prod.apps.6a28.example.opentlc.com

OpenShift Tasks Demo

Home

Logger

Log Info

Log Warning

Log Error

Load Generator

Seconds

Load!

Danger Zone

HEALTHY

Toggle Health

Kill Instance

Info

Pod Hostname	tasks-3-x85c6
Pod IP	null
Used Memory	1250 MB
Session ID	uQfx-wOe7IJMX4loqm7S-ConNE4PFxguGM6AAAd_e

Messages

Nothing to report.

Nodejs-mongo-persistent

nodejs-mongo-persistent-smoke-test.apps.6a28.example.opentlc.com

Welcome to your Node.js application on OpenShift

How to use this example application

For instructions on how to use this application with OpenShift, start by reading the [Developer Guide](#).

Deploying code changes

The source code for this application is available to be forked from the [OpenShift GitHub repository](#). You can configure a webhook in your repository to make OpenShift automatically start a build whenever you push your code:

1. From the Web Console homepage, navigate to your project
2. Click on Browse > Builds
3. Click the link with your BuildConfig name
4. Click the Configuration tab
5. Click the "Copy to clipboard" icon to the right of the "GitHub webhook URL" field
6. Navigate to your repository on GitHub and click on repository settings > webhooks > Add webhook
7. Paste your webhook URL provided by OpenShift in the "Payload URL" field
8. Change the "Content type" to 'application/json'
9. Leave the defaults for the remaining fields — that's it!

After you save your webhook, if you refresh your settings page you can see the status of the ping that Github sent to OpenShift to verify it can reach the server.

Note: adding a webhook requires your OpenShift server to be reachable from GitHub.

Working in your local Git repository

If you forked the application from the OpenShift GitHub example, you'll need to manually clone the repository to your local system. Copy the application's source code Git URL and then run:

Managing your application

Documentation on how to manage your application from the Web Console or Command Line is available at the [Developer Guide](#).

Web Console

You can use the Web Console to view the state of your application components and launch new builds.

Command Line

With the [OpenShift command line interface](#) (CLI), you can create applications and manage projects from a terminal.

Development Resources

- [OpenShift Documentation](#)
- [Openshift Origin GitHub](#)
- [Source To Image GitHub](#)
- [Getting Started with Node.js on OpenShift](#)
- [Stack Overflow questions for OpenShift](#)
- [Git documentation](#)

Request information

Page view count: 389

DB Connection Info:

Type: MongoDB
URL: mongodb://172.30.154.104:27017/sampledb

Git Bash

```
PLAY RECAP *****
infranode1.6a28.internal : ok=176 changed=80 unreachable=0 failed=0
infranode2.6a28.internal : ok=176 changed=80 unreachable=0 failed=0
loadbalancer1.6a28.internal : ok=60 changed=17 unreachable=0 failed=0
localhost                : ok=66 changed=24 unreachable=0 failed=0
master1.6a28.internal    : ok=1266 changed=488 unreachable=0 failed=0
master2.6a28.internal    : ok=362 changed=165 unreachable=0 failed=0
master3.6a28.internal    : ok=362 changed=165 unreachable=0 failed=0
node1.6a28.internal      : ok=175 changed=79 unreachable=0 failed=0
node2.6a28.internal      : ok=175 changed=79 unreachable=0 failed=0
node3.6a28.internal      : ok=175 changed=79 unreachable=0 failed=0
support1.6a28.internal   : ok=60 changed=14 unreachable=0 failed=0

INSTALLER STATUS *****
Initialization           : Complete (0:00:59)
Health Check             : Complete (0:00:18)
Node Bootstrap Preparation : Complete (0:05:58)
etcd Install             : Complete (0:01:12)
NFS Install              : Complete (0:00:14)
Load Balancer Install    : Complete (0:00:20)
Master Install           : Complete (0:05:32)
Master Additional Install : Complete (0:00:59)
Node Join                : Complete (0:00:47)
Hosted Install           : Complete (0:01:06)

The use of NFS for the core OpenShift Container Platform components is not recommended, as NFS (and the NFS Protocol) does not provide the
proper consistency needed for the applications that make up the OpenShift Container Platform infrastructure.
Cluster Monitoring Operator : Complete (0:01:18)
Web Console Install        : Complete (0:00:24)
Console Install            : Complete (0:00:26)
Metrics Install            : Complete (0:02:25)
metrics-server Install     : Complete (0:00:53)
Logging Install            : Complete (0:03:24)
Prometheus Install         : Complete (0:00:45)
Service Catalog Install    : Complete (0:01:50)
[root@bastion ~]# exit
logout
[djordje.gajisin-devoteam.com@bastion ~]$
```