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/dulecux/ocp_advanced_deployment_homework

Step	Description	Command
1.	Go to root	sudo -i
2.	Clone the git repository	git clone https://github.com/dulecux/ocp_advanced_deployment_homework.git
3.	Run the ansible playbook	ansible-playbook ./ocp_advanced_deployment_homework/homework.yaml
4.	Uninstall the cluster (optional)	sh ./ocp_advanced_deployment_homework/scripts/uninstall.sh

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 provides 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.750d.example.opentlc.com	gogs/gogs
jenkins	jenkins-tasks-dev.apps.750d.example.opentlc.com	andrew/r3dh4t1!
nexus	nexus-tasks-dev.apps.750d.example.opentlc.com	admin/admin123
sonarqube	sonarqube-tasks-dev.apps.750d.example.opentlc.com	admin/admin
tasks	tasks-tasks-prod.apps.750d.example.opentlc.com	
node-js-app	nodejs-mongo-persistent-smoke-test.apps.750d.example.opentlc.com	

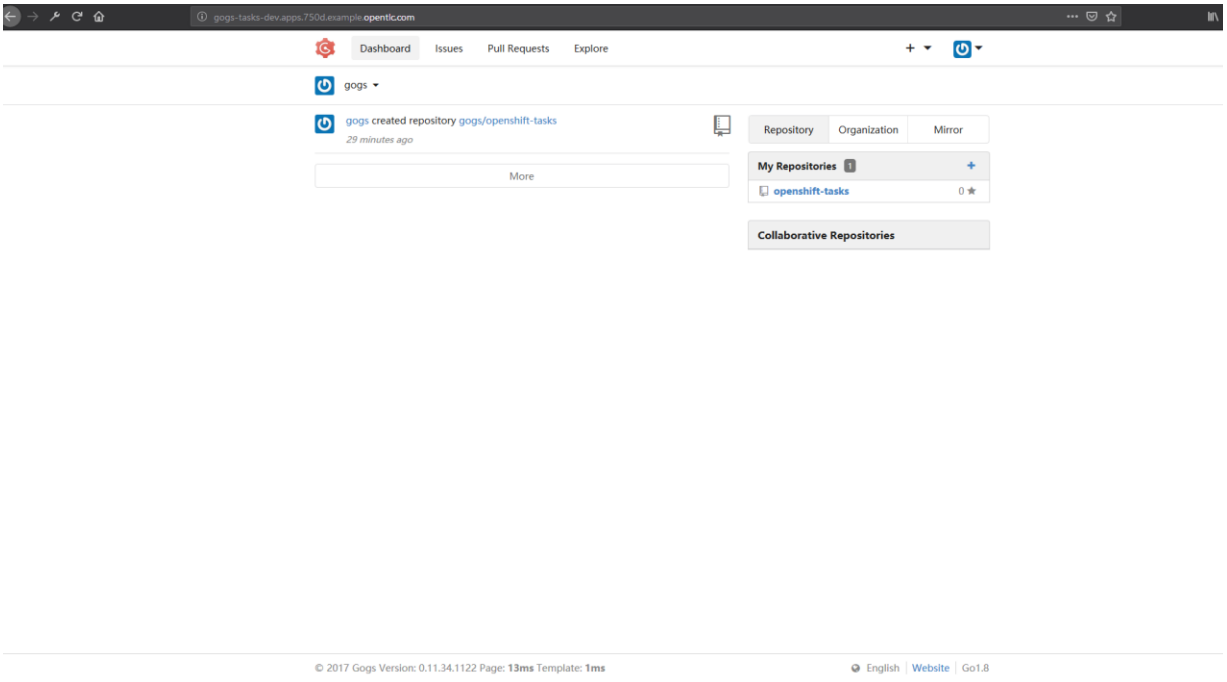
Release version – 3.10.34

Instructor – Jindrich Kana

Venue – Garni Hotel Centar, Novi Sad Serbia

Participant – Dusan Culibrk (dusan.culibrk@devoteam.com)

Gogs



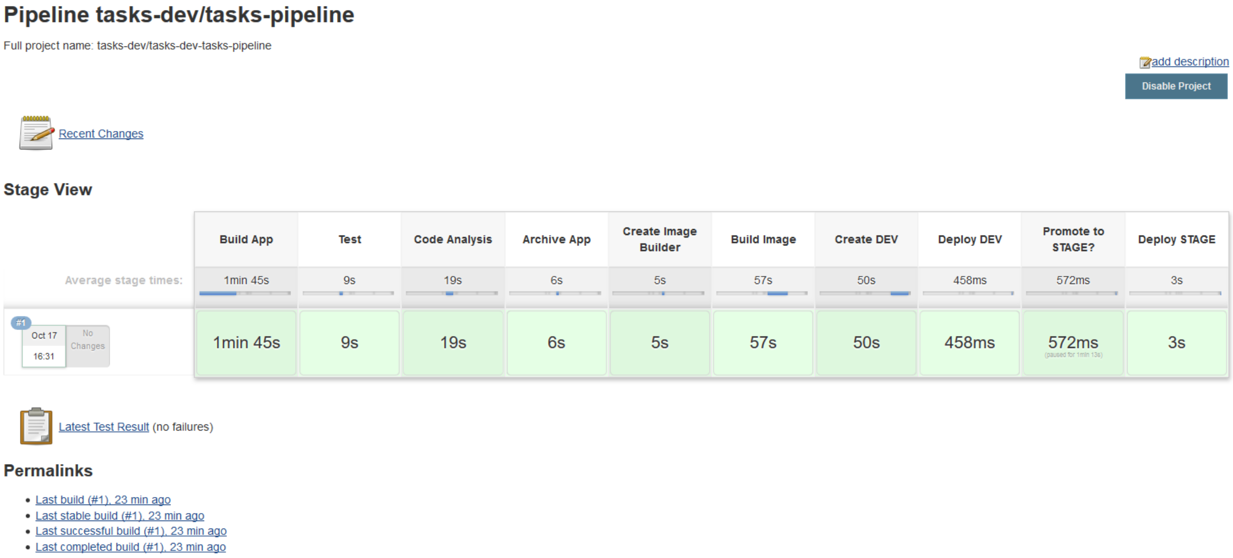
gogs-tasks-dev.apps.750d.example.opentlc.com

gogs created repository gogs/openshift-tasks

29 minutes ago

© 2017 Gogs Version: 0.11.34.1122 Page: **13ms** Template: **1ms**

Jenkins



Pipeline tasks-dev/tasks-pipeline

Full project name: tasks-dev/tasks-dev-tasks-pipeline

Stage View

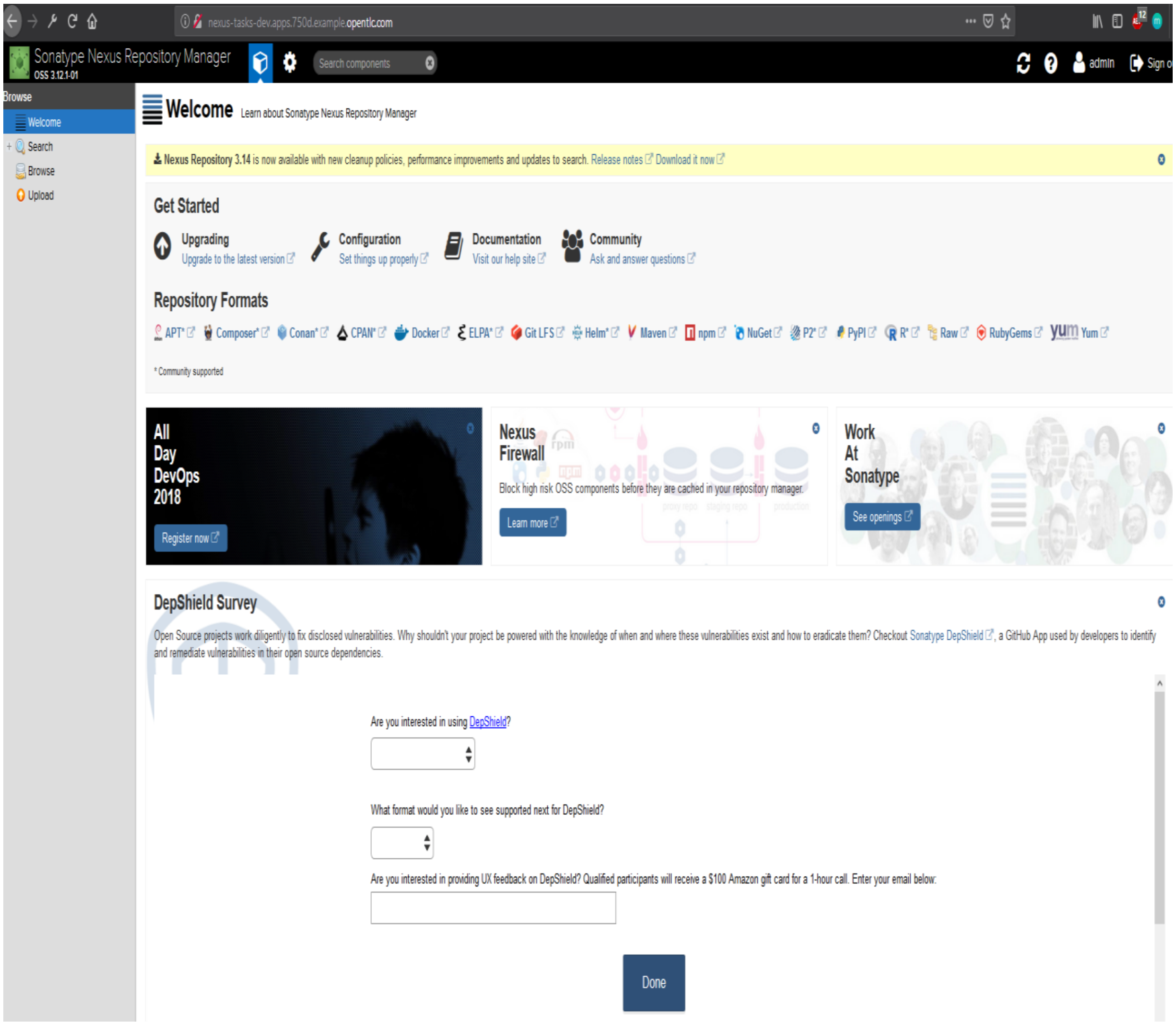
	Build App	Test	Code Analysis	Archive App	Create Image Builder	Build Image	Create DEV	Deploy DEV	Promote to STAGE?	Deploy STAGE
Average stage times:	1min 45s	9s	19s	6s	5s	57s	50s	458ms	572ms	3s
#1 Oct 17 16:31 No Changes	1min 45s	9s	19s	6s	5s	57s	50s	458ms	572ms (paused for 1min 13s)	3s

Latest Test Result (no failures)

Permalinks

- Last build (#1), 23 min ago
- Last stable build (#1), 23 min ago
- Last successful build (#1), 23 min ago
- Last completed build (#1), 23 min ago

Nexus



nexus-tasks-dev.apps.750d.example.opentlc.com
Sonatype Nexus Repository Manager
OSS 3.12.1-01
Search components
admin
Sign o
Browse
Welcome
Search
Browse
Upload

Welcome Learn about Sonatype Nexus Repository Manager

Nexus Repository 3.14 is now available with new cleanup policies, performance improvements and updates to search. Release notes Download it now

Get Started

Upgrading
Upgrade to the latest version

Configuration
Set things up properly

Documentation
Visit our help site

Community
Ask and answer questions

Repository Formats

APT* Composer* Conan* CPAN* Docker ELPA* Git LFS Helm* Maven npm NuGet P2* PyPI R* Raw RubyGems yum Yum

* Community supported

All Day DevOps 2018
Register now

Nexus Firewall
Block high risk OSS components before they are cached in your repository manager.
Learn more

Work At Sonatype
See openings

DepShield Survey

Open Source projects work diligently to fix disclosed vulnerabilities. Why shouldn't your project be powered with the knowledge of when and where these vulnerabilities exist and how to eradicate them? Checkout Sonatype DepShield, a GitHub App used by developers to identify and remediate vulnerabilities in their open source dependencies.

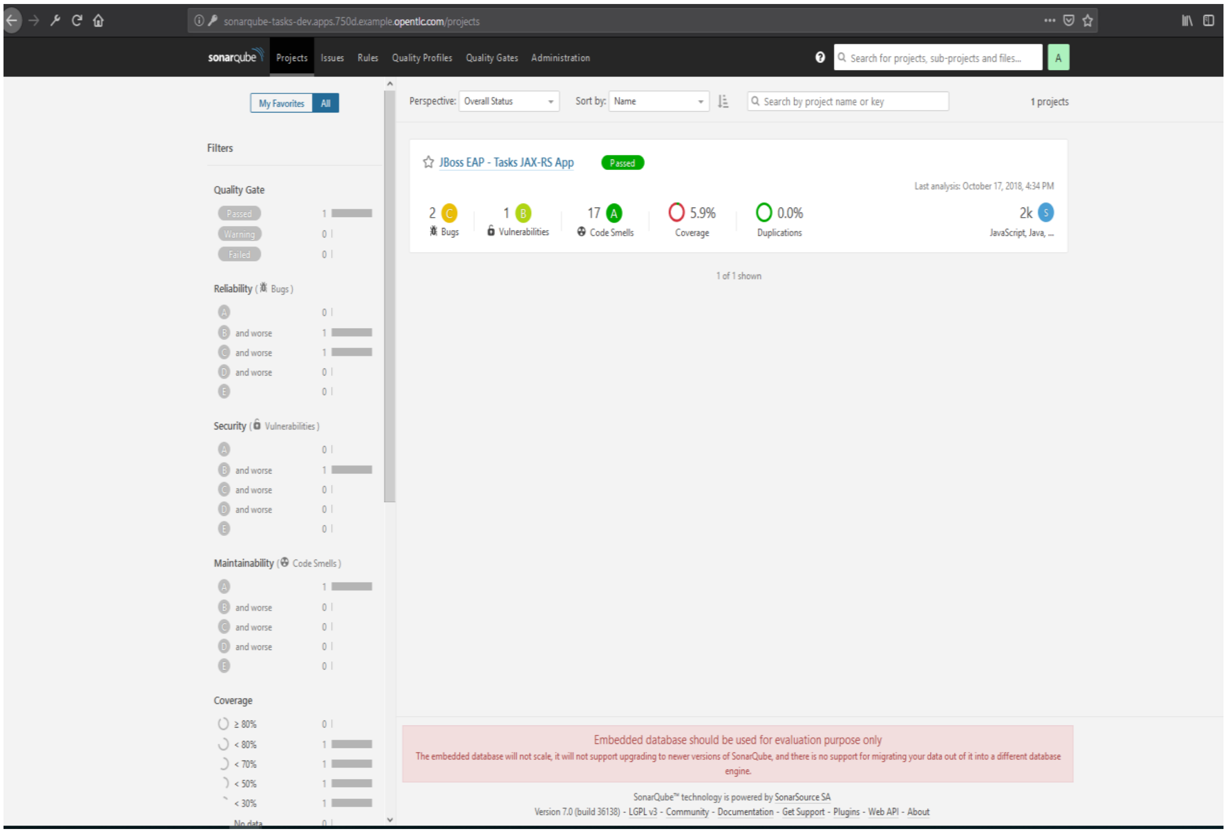
Are you interested in using DepShield?

What format would you like to see supported next for DepShield?

Are you interested in providing UX feedback on DepShield? Qualified participants will receive a $100 Amazon gift card for a 1-hour call. Enter your email below:

Done

Sonarqube



JBoss EAP - Tasks JAX-RS App Passed

Last analysis: October 17, 2018, 4:34 PM

2 C	1 B	17 A	5.9%	0.0%	2k S
Bugs	Vulnerabilities	Code Smells	Coverage	Duplications	JavaScript, Java, ...

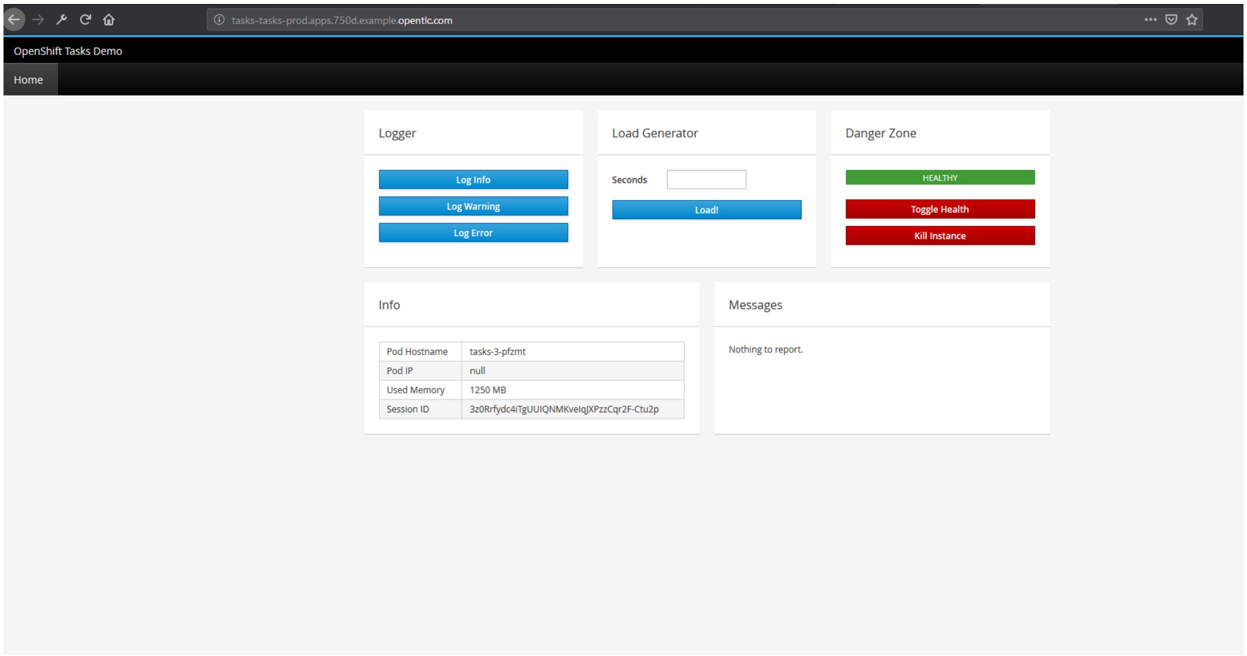
Embedded database should be used for evaluation purpose only

The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA

Version 7.0 (build 36138) - LGPL v3 - Community - Documentation - Get Support - Plugins - Web API - About

Task-prod



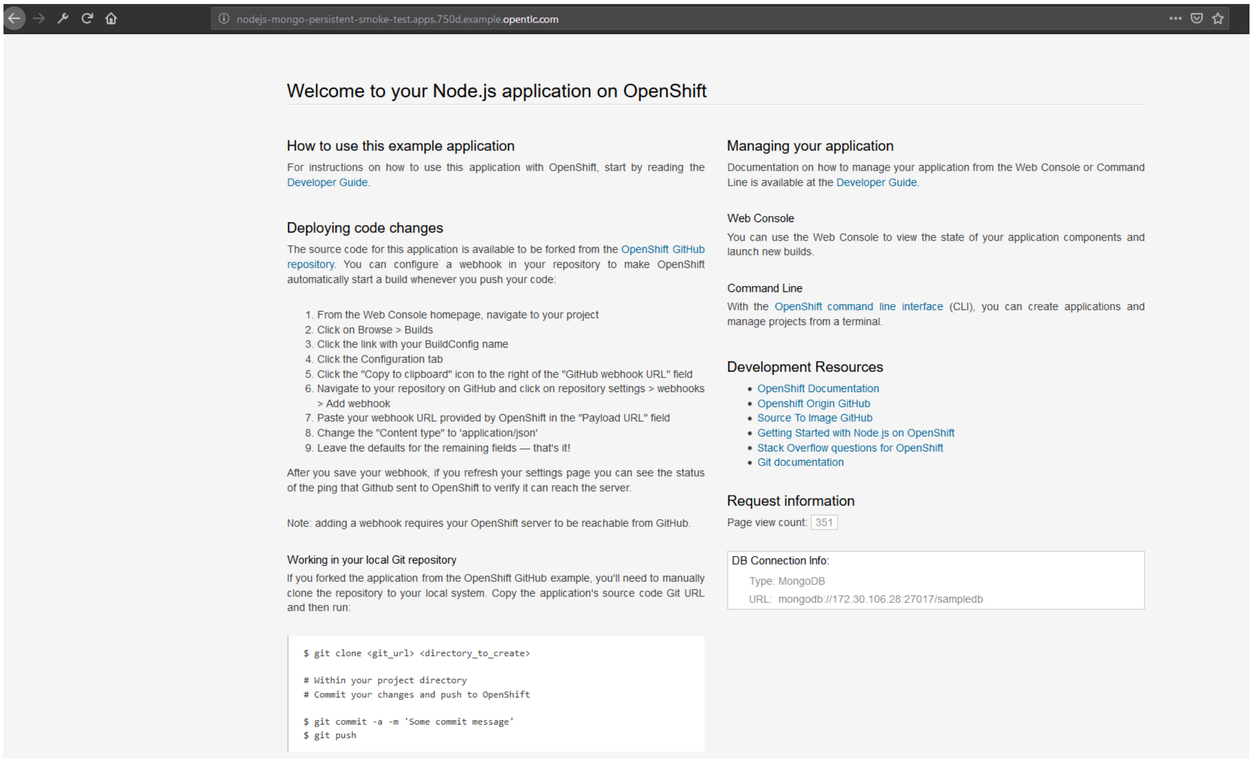
Info

Pod Hostname	tasks-3-pfzmt
Pod IP	null
Used Memory	1250 MB
Session ID	3z0Rrfydc4iTgUUIQNMKveIqJXPzzCqr2F-Ctu2p

Messages

Nothing to report.

Nodejs-mongo-persistent



nodejs-mongo-persistent-smoke-test.apps.750d.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:

```
$ git clone <git_url> <directory_to_create>

# Within your project directory
# Commit your changes and push to OpenShift

$ git commit -a -m 'Some commit message'
$ git push
```

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

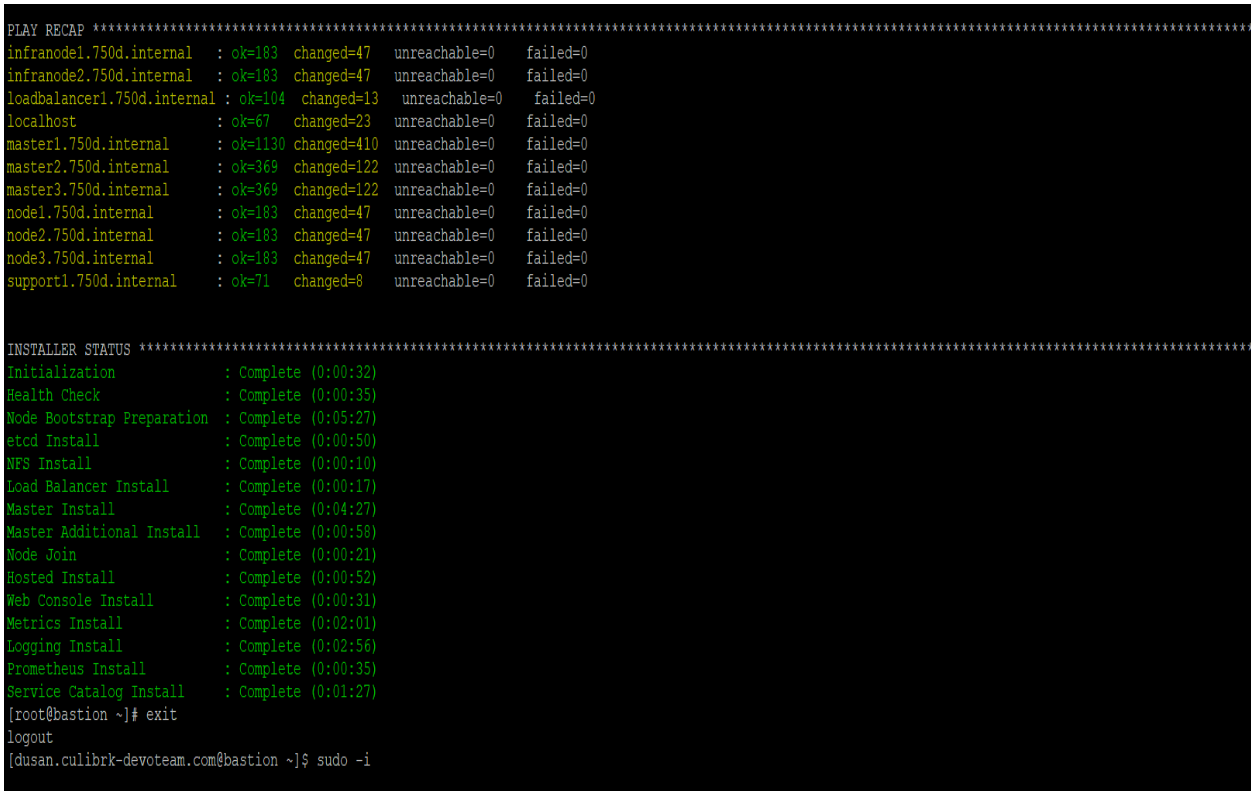
Page view count: 351

DB Connection Info:

Type: MongoDB

URL: mongodb://172.30.106.28:27017/sampledb

Putty



```
PLAY RECAP ******************************************************************************************************************************************************
infranode1.750d.internal   : ok=183  changed=47   unreachable=0    failed=0
infranode2.750d.internal   : ok=183  changed=47   unreachable=0    failed=0
loadbalancer1.750d.internal : ok=104  changed=13   unreachable=0    failed=0
localhost                  : ok=67   changed=23   unreachable=0    failed=0
master1.750d.internal      : ok=1130 changed=410  unreachable=0    failed=0
master2.750d.internal      : ok=369  changed=122  unreachable=0    failed=0
master3.750d.internal      : ok=369  changed=122  unreachable=0    failed=0
node1.750d.internal        : ok=183  changed=47   unreachable=0    failed=0
node2.750d.internal        : ok=183  changed=47   unreachable=0    failed=0
node3.750d.internal        : ok=183  changed=47   unreachable=0    failed=0
support1.750d.internal     : ok=71   changed=8    unreachable=0    failed=0


INSTALLER STATUS ************************************************************************************************************************************************
Initialization              : Complete (0:00:32)
Health Check                : Complete (0:00:35)
Node Bootstrap Preparation  : Complete (0:05:27)
etcd Install                : Complete (0:00:50)
NFS Install                 : Complete (0:00:10)
Load Balancer Install       : Complete (0:00:17)
Master Install              : Complete (0:04:27)
Master Additional Install   : Complete (0:00:58)
Node Join                   : Complete (0:00:21)
Hosted Install              : Complete (0:00:52)
Web Console Install         : Complete (0:00:31)
Metrics Install             : Complete (0:02:01)
Logging Install             : Complete (0:02:56)
Prometheus Install          : Complete (0:00:35)
Service Catalog Install     : Complete (0:01:27)
[root@bastion ~]# exit
logout
[dusan.culibrk-devoteam.com@bastion ~]$ sudo -i
```