

A multi-nodes config using VirtualBox to showcase  
**end-to-end DevOps using**  
**OpenShift Container Platform**

Gabriel Bechara

Principal Solution Architect @ Red Hat

<https://github.com/gbechara/osedevops>

# About this showcase

- The installer will
  - create multiple vagrant machines : ose-master, ose-node-x, ose-infra
  - configure those machines to add the pre-requisites using a Ansible script
  - install OpenShift Container Platform 3.x calling the provided OpenShift Ansible installer
  - configure and install
    - registry
    - router
    - metrics
    - gitlab
    - Nexus (for java and nodejs offline build)
    - Examples : offline templates and pipelines

# Architecture

## DNS entry for each VM

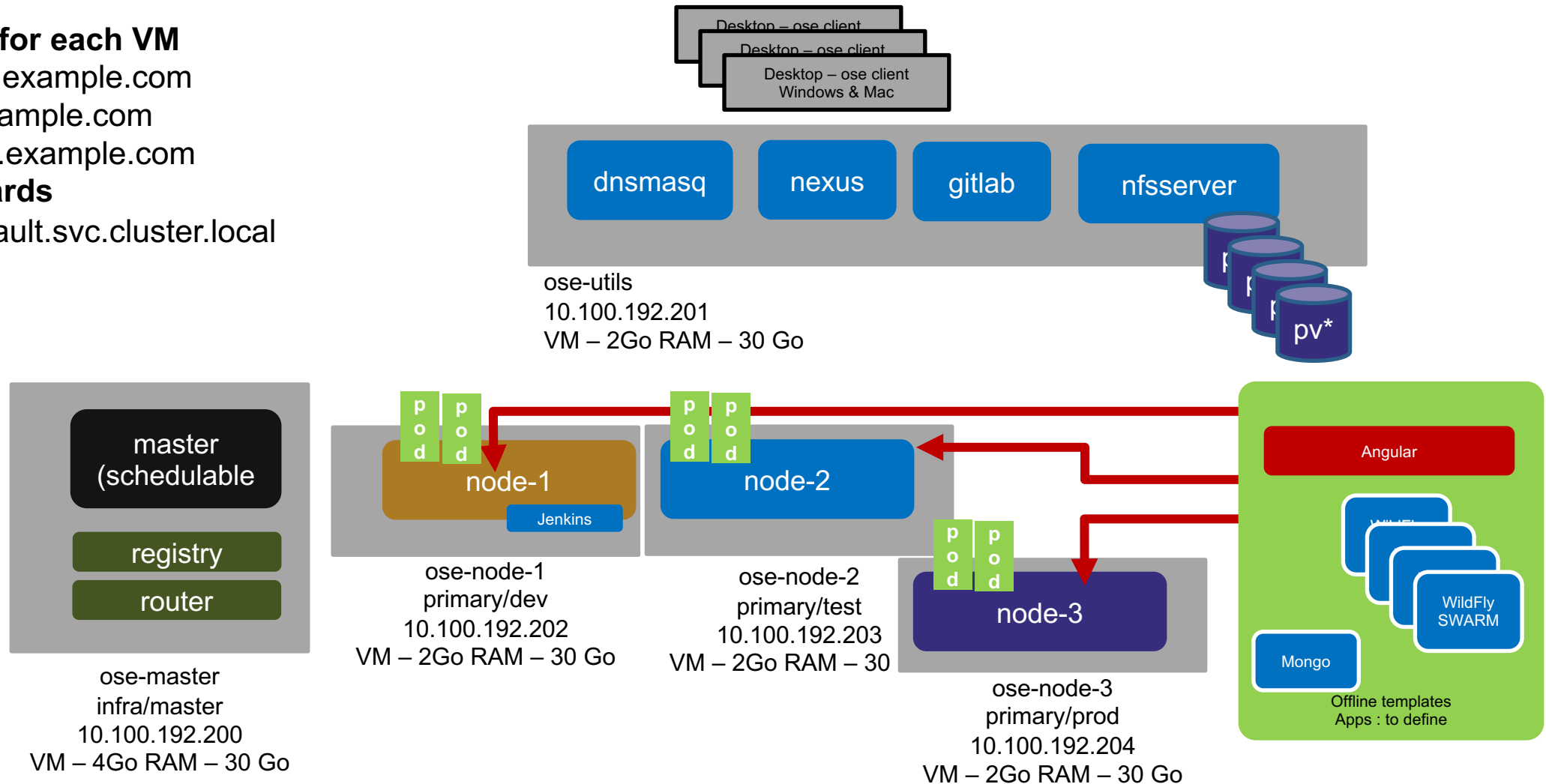
ose-master.example.com

ose-utils.example.com

ose-node-1.example.com

## dns Wildcards

\*.router.default.svc.cluster.local



# Prerequisites

- Virtual Box (tested on Version 5.0.20 r106931)
- Vagrant (tested on 1.8.1)
- Valid RHEL7 & OpenShift Container Platform Subscription
  - Trials are available on <https://www.openshift.com/container-platform/trial.html>
- Create your own Vagrant Box
  - Create a VirtualBox with RHEL7 “Server”
  - Add to this VirtualBox your valid Subs OR use vagrant-registration plugin
  - Convert this Box into a Vagrant Base Box located on your local disk

Instruction are here <https://github.com/gbechara/osdevops/blob/master/create-vagrant-base-box.md>

Notes :

name this base box rhel72-server-base.box (or change the name in the Vagrantfile)

do not share this box : it may contain your subscriptions

you may need to load update your box (to update the cache in ~/.vagrant.d/boxes)

```
# vagrant box remove rhel72-server-base.box
```

```
# vagrant box add rhel72-server-base.box --force --name rhel72-server-base.box
```

# Usage (1/2)











- Get the source code
    - > git clone <https://github.com/gbechara/osedevops.git>
  - If your Sub is not in the Vagrant box you can use the vagrant-registration plugin
    - add in ~/.vagrant.d/Vagrantfile or in the current Vagrantfile (you got from github) the following

```
Vagrant.configure('2') do |config|  
  config.registration.username = '<your Red Hat username>  
  config.registration.password = '<your Red Hat password>  
  config.registration.pools = [ 'thepoolthatcontainstheadequatesubs' ]  
End
```
- Launch
  - > Change the passwords in the file Vagrantfile to match the root password of your box
  - This password is needed to copy the generated ssh key to all OpenShift nodes
  - > cd osedevops
  - > # vagrant plugin install vagrant-cachier # not recommended some error related to yum update may occur using vagrant-cachier you may want not install this plug-in
  - > vagrant plugin install vagrant-registration # (if your sub is not in the vbox)
  - > vagrant up
- configure your host to add a dnsserver
  - on linux add in /etc/resolv.conf









```
search example.com  
10.100.192.201
```
- on osx create 2 files named 'example.com' and 'router.default.svc.cluster.local' in /etc/resolver add to those files
- ```
nameserver 10.100.192.201
```




# Usage (2/2)

- Openshift Web Console
  - <https://ose-master.example.com:8443/console/>
  - User dev1/dev1 have access to the development project
  - User test1/test1 have access to development, testing, ci and production (when created) project
- Jenkins
  - <https://jenkins-cicd.router.default.svc.cluster.local>
  - Jenkins user is admin/password
  - 3 preconfigured pipelines are provided
    - **pipeline-development-nodejs**
    - **pipeline-development-ticket-monster**
    - **pipeline-development-ticket-cake-php**
- GitLab
  - <http://gitlab.example.com/>
  - User gabriel/weareawesome
  - 3 projects used by offline templates and one for jenkins are pre-populated on the local gitlab

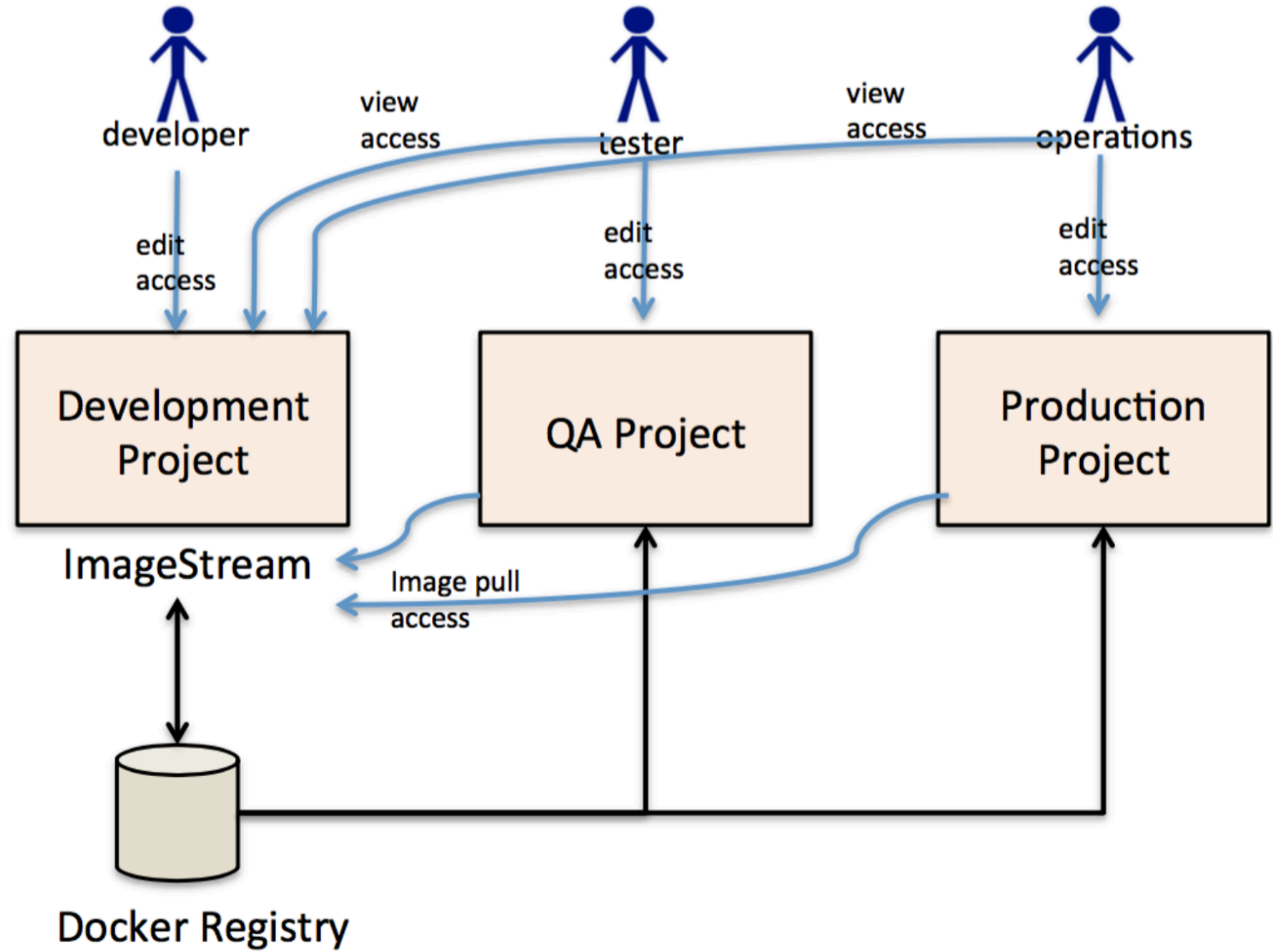
| All pipeline +                                                                      |                                                                                     |                                                     |
|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|-----------------------------------------------------|
| S                                                                                   | M                                                                                   | Nom du projet ↓                                     |
|  |  | <a href="#">build-development-ticket-monster</a>    |
|  |  | <a href="#">deploy-testing-ticket-monster</a>       |
|  |  | <a href="#">pipeline-development-cake-php</a>       |
|  |  | <a href="#">pipeline-development-nodejs</a>         |
|  |  | <a href="#">pipeline-development-ticket-monster</a> |

Icône: [S](#) [M](#) [L](#) [Légende](#)

| Instant Apps                                                                         |                                                                                                    | <a href="#">See all</a>                                                                                                                                                   |
|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | <b>1-cakephp-offline</b><br>INSTANT-APP QUICKSTART PHP CAKEPHP                                     |     |
|   | <b>1-eap64-ticket-monster-offline</b><br>Version: 1.2.0<br>INSTANT-APP EAP JAVAEE JAVA JBOSS XPAAS |                                                                                        |
|  | <b>1-nodejs-offline-example</b><br>INSTANT-APP QUICKSTART NODEJS                                   |   |

| Signed in successfully.                                                               |                                     |
|---------------------------------------------------------------------------------------|-------------------------------------|
| Your Projects Starred Projects Explore Projects                                       |                                     |
|  | Gabriel Bechara / nodejs-example    |
|  | Gabriel Bechara / osedevops-jenkins |
|  | Gabriel Bechara / cake-php          |


# Example



Based on <https://blog.openshift.com/promoting-applications-across-environments/>

# Jenkins Pipeline : from dev to production

→ Approvals between environments and creation of the next env if it does not exist

**Jenkins**

rechercher

Jenkins Admin | se déconnecter

Jenkins » pipeline-development-ticket-monster » [Rafraîchissement automatique](#)

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Lancer un build](#)

[Supprimer Pipeline](#)


[Configurer](#)

[Move](#)

[Full Stage View](#)

**Pipeline pipeline-development-ticket-monster**

[Ajouter une description](#)

 [Recent Changes](#)

**Stage View**

**Historique des builds** [tendance](#)

find x

#43 juin 2016 23:04

#33 juin 2016 23:02

#23 juin 2016 22:42

#13 juin 2016 22:36

[RSS des builds](#) [RSS des échecs](#)

Average stage times:  
(Average full run time: ~25s)

#4Jun 04 01:04No Changes

#3Jun 04 01:02No Changes

#2

| Build image and deploy in Dev | Wait for approval                   | Deploy to testing   | Wait for approval                  | Deploy to production |
|-------------------------------|-------------------------------------|---------------------|------------------------------------|----------------------|
| 8s                            | 714ms                               | 5s                  | NaNy NaNd                          | 8s                   |
| 8s<br>master                  | 471ms<br>(paused for 46s)<br>master | 7s<br>master        | 263ms<br>(paused for 8s)<br>master | 8s<br>master         |
| 7s<br>master                  | 418ms<br>(paused for 7s)<br>master  | 6s<br>master failed |                                    |                      |
|                               |                                     |                     |                                    |                      |



# Notes

- Installation works only when connected to internet
  - It may take time, around 40 minutes, depending on the roles you add
  - The installation may not work when having network issue or if your laptop suspends during the install
- 3 templates can then be used to do offline demos
- During the install the sample gitlab, nexus, the docker images, the jenkins plugins are all populated to work offline for the all templates
- 3 users (gabriel, dev1, test1), 2 projects (development, testing) are created to deploy the sample, the jenkins job “pipeline-development-ticket-monster” will create a third project

# Roadmap

- ✓ Add NFS for persistent Volumes
- ✓ Add a git server with 3 pre-populated samples
- ✓ Add a nexus with proxies configured for Java and NodeJS samples
- ✓ Add templates for offline demos using the git server and the nexus
- ✓ Deploy and example to populate the docker registry, gitlab and nexus (for offline demos)
- ✓ Showcase an application promotion across environments :
  - ✓ preconfigured pipelines for the 3 samples offline JBoss, NodeJS and PHP Projects
- ✓ Add a integrated Dev Pipeline in the Dev project (OCP3.3)
- Add other samples
  - Deploying WildFly SWARM microservices on OpenShift
  - Covering the entire DevOps lifecycle (adding Jira ?)
  - Other ideas ?
  - ....

