Achieving CI/CD with Kubernetes

Ramit Surana

@ramitsurana

/in/ramitsurana

Agenda

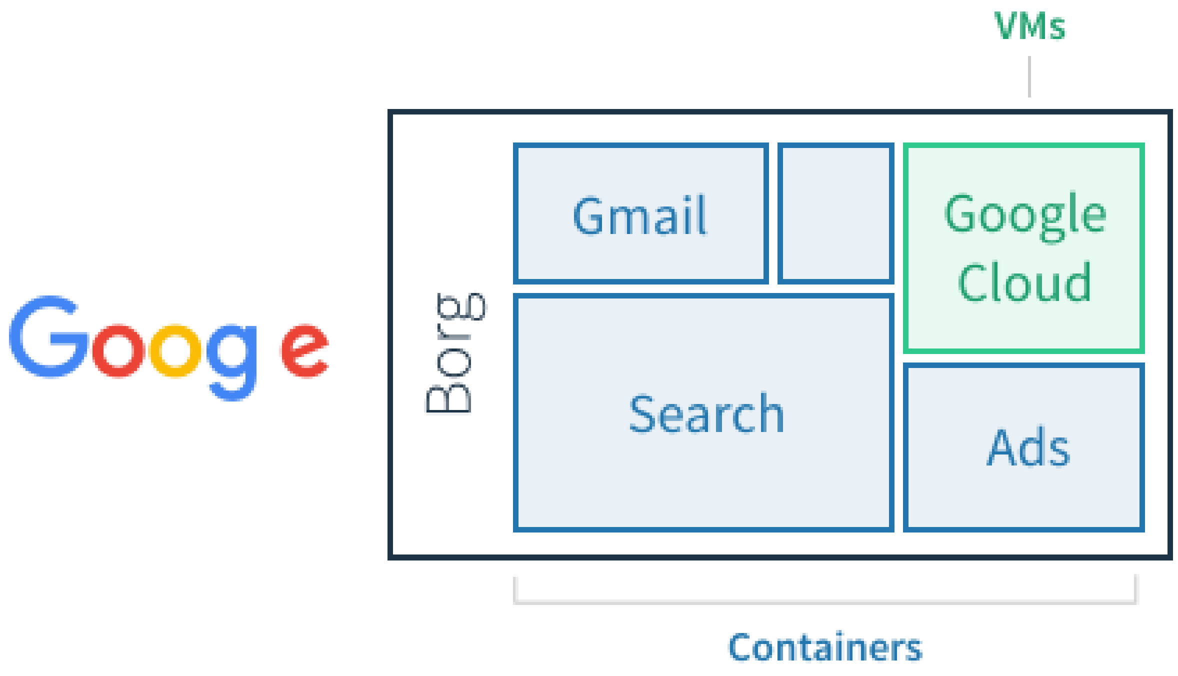
- History of Kubernetes
- The evolution
- How google uses it ?
- About Jenkins
- How to configure it ?
- Fabric8
- How to configure it ?
- The Final Picture
- Other Tools
- Awesome Kubernetes
- Future of Kubernetes

About Me

- Open Source Guy
- Contributor to Kubernetes community
- Author of awesome-kubernetes
- ramitsurana@gmail.com

History of kubernetes

- Earlier known as Borg
- Borg -> Omega -> Kubernetes
- Google used MPM (Midas Package Manager) to build and deploy container images.



How Google does it?

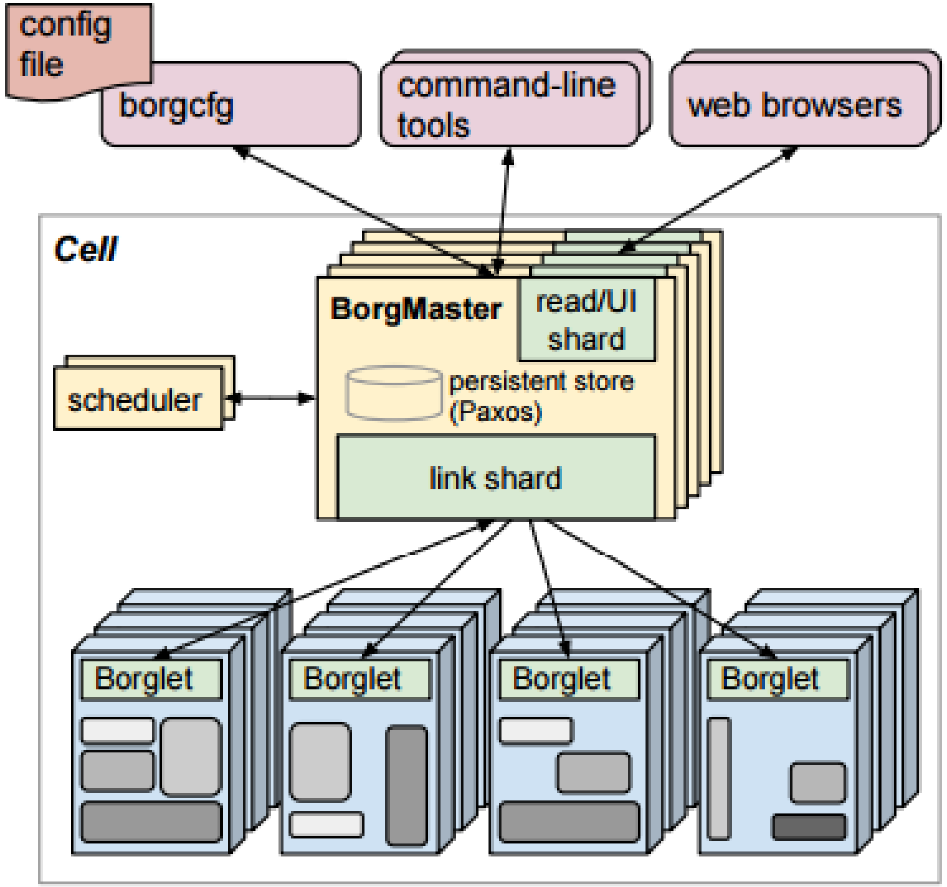
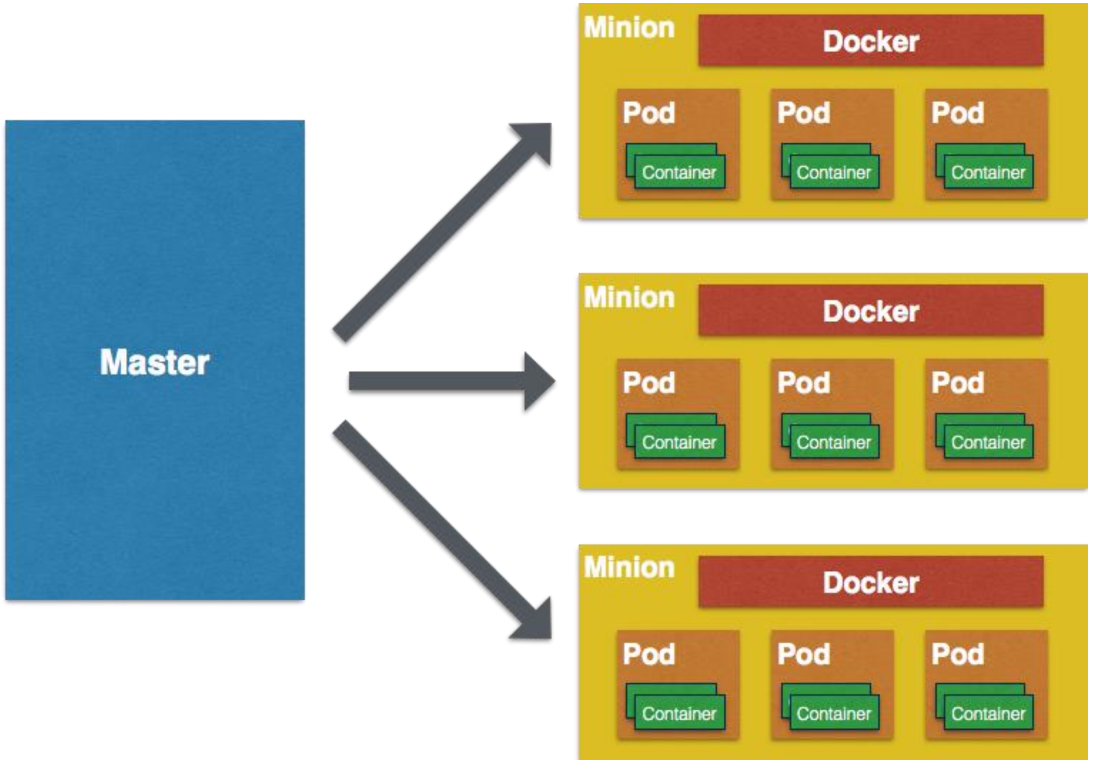


Figure 1: The high-level architecture of Borg. *Only a tiny fraction of the thousands of worker nodes are shown.*

Evolution of Kubernetes

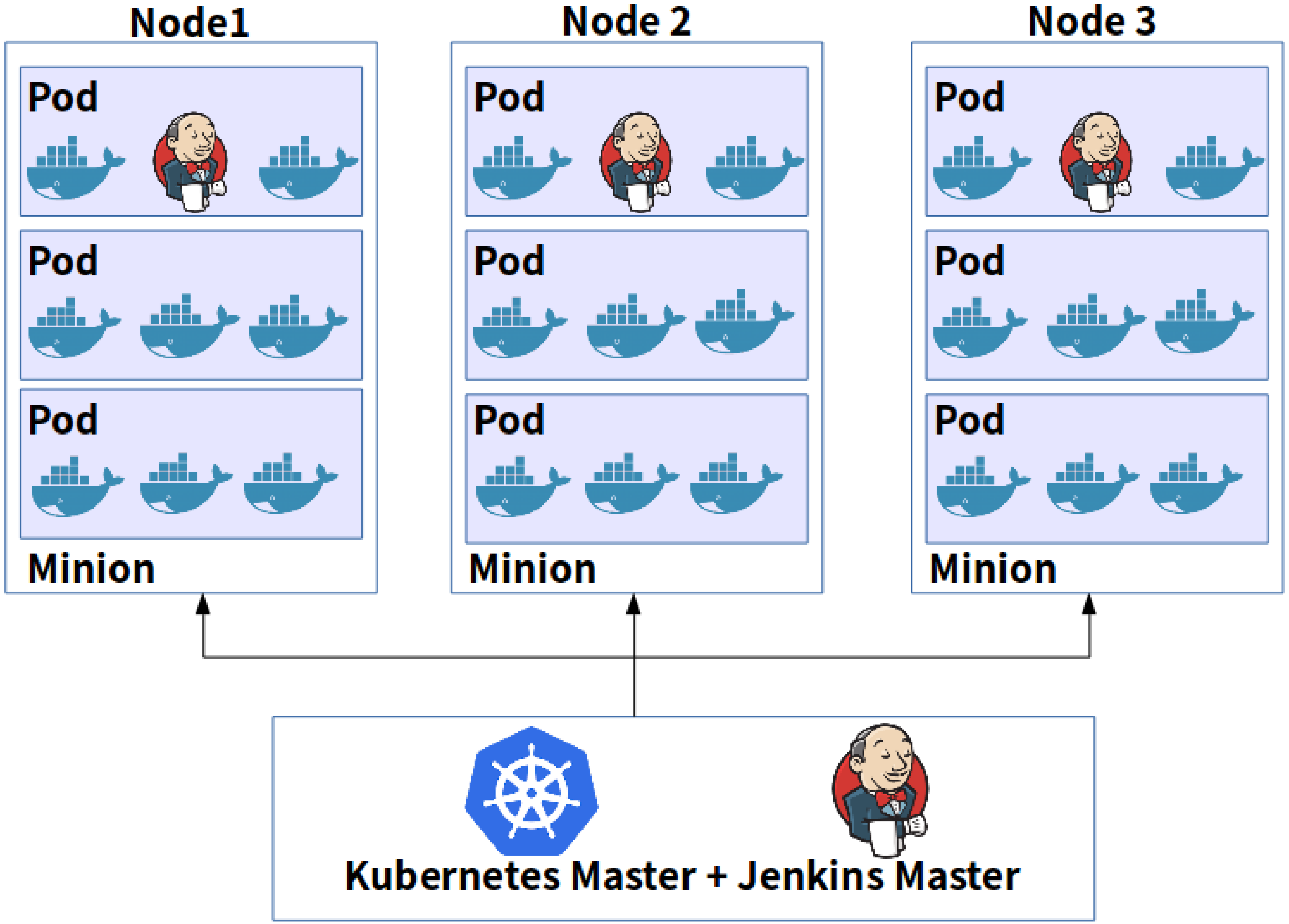
- Benefits of containerization go beyond merely enabling higher levels of utilization.
- Need for an orchestration engine with capablities.
- Need of a stable model,to use docker at large scale that could run on several cloud platforms.



About Jenkins

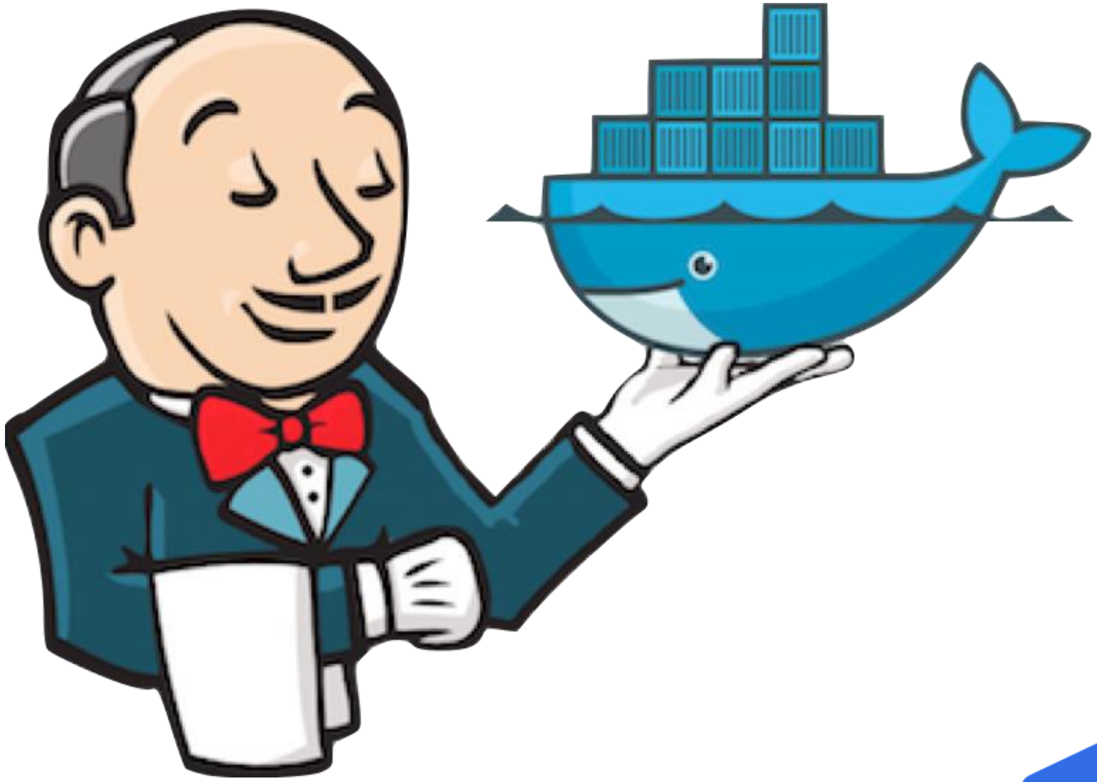
- Earlier known as Hudson.
- Primarily a java-based program
- Leading open source CI server
- Used by leading companies such as Github,Netflix,Linkedin etc.

Workflow with Docker



How to configure it ?

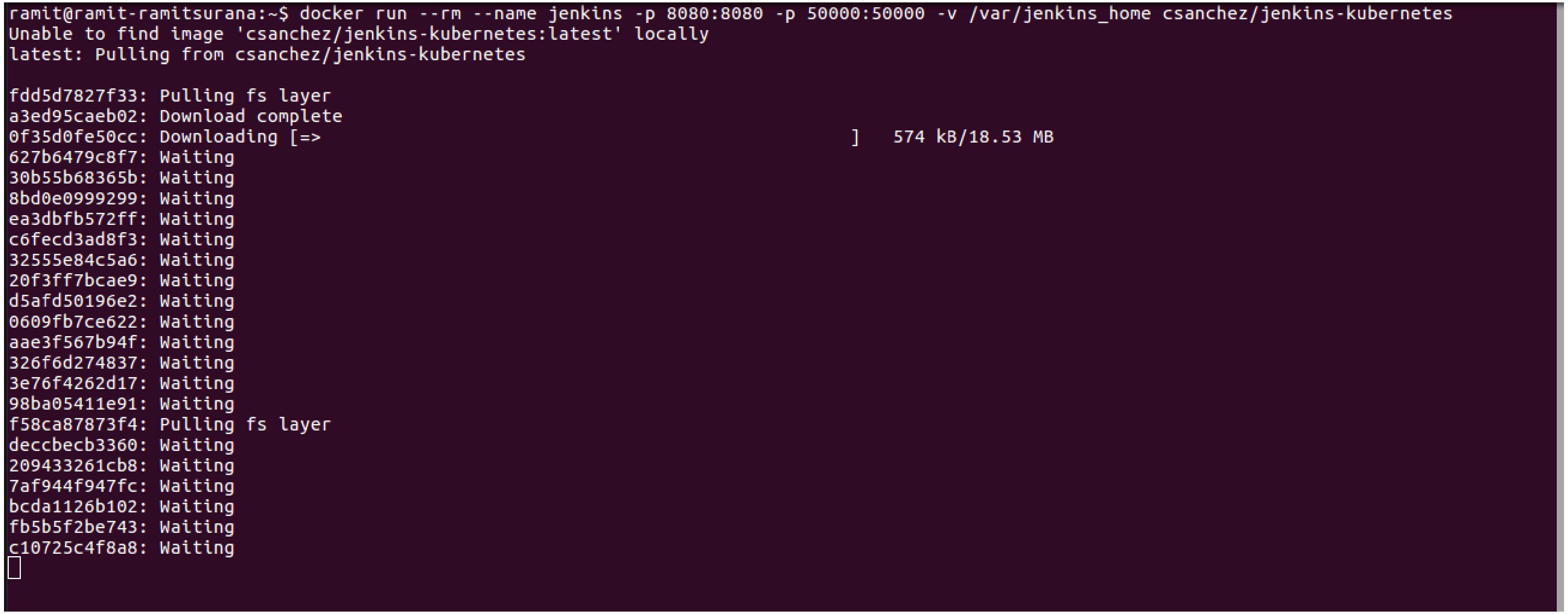
- Find the k8s-jenkins plugin at https://github.com/jenkinsci/kubernetes-plugin
- Works using JNLP(Java Network Launch Protocol)



How to configure it ?

- Running the Jenkins image

docker run –rm –name jenkins – p 8080:8080 – p 50000:50000 –v /var/jenkins_home csanchez/jenkins-kubernetes



```
ramit@ramit-ramitsurana:~$ docker run --rm --name jenkins -p 8080:8080 -p 50000:50000 -v /var/jenkins_home csanchez/jenkins-kubernetes
Unable to find image 'csanchez/jenkins-kubernetes:latest' locally
latest: Pulling from csanchez/jenkins-kubernetes

fdd5d7827f33: Pulling fs layer
a3ed95caeb02: Download complete
0f35d0fe50cc: Downloading [=>                                                ]   574 kB/18.53 MB
627b6479c8f7: Waiting
30b55b68365b: Waiting
8bd0e0999299: Waiting
ea3dbfb572ff: Waiting
c6fecd3ad8f3: Waiting
32555e84c5a6: Waiting
20f3ff7bcae9: Waiting
d5afd50196e2: Waiting
0609fb7ce622: Waiting
aae3f567b94f: Waiting
326f6d274837: Waiting
3e76f4262d17: Waiting
98ba05411e91: Waiting
f58ca87873f4: Pulling fs layer
deccbecb3360: Waiting
209433261cb8: Waiting
7af944f947fc: Waiting
bcda1126b102: Waiting
fb5b5f2be743: Waiting
c10725c4f8a8: Waiting
```

How to configure it ?

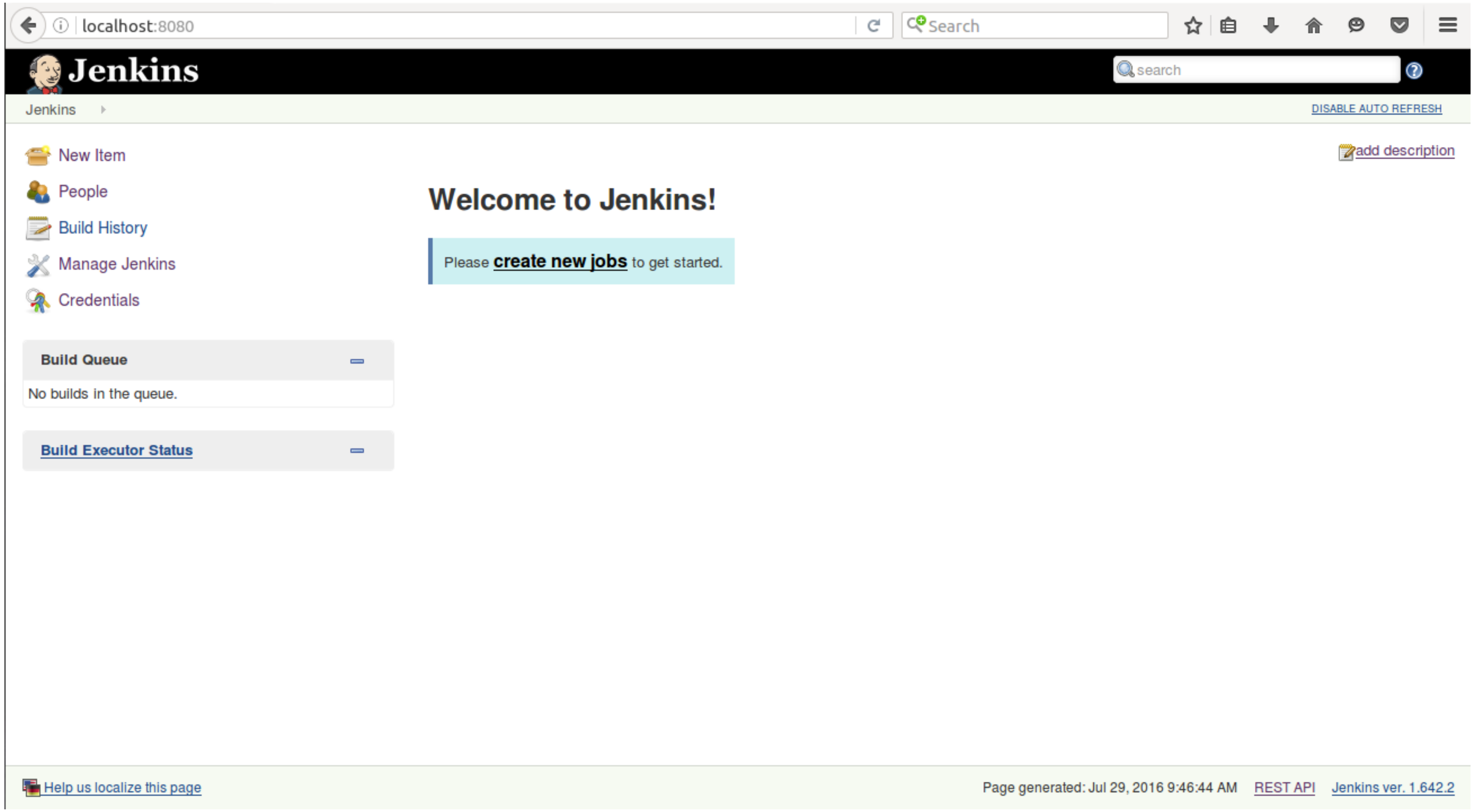
- Problem
- It works only one time
- Because of stateless containers the data gets erased after a restart.
- A better solution is to use a container to store and use data to run the jenkins container
- Better Solution

```
docker create --name jenkins-k8s csanchez/jenkins-kubernetes
```

```
docker run --volumes-from jenkins-k8s -p 8080:8080 -p 50000:50000 -v /var/jenkins_home csanchez/jenkins-kubernetes
```

How to configure it ?

- Checkout http://localhost:8080 in your browser



Welcome to Jenkins!

Please **create new jobs** to get started.

Build Queue

No builds in the queue.

Build Executor Status

Help us localize this page

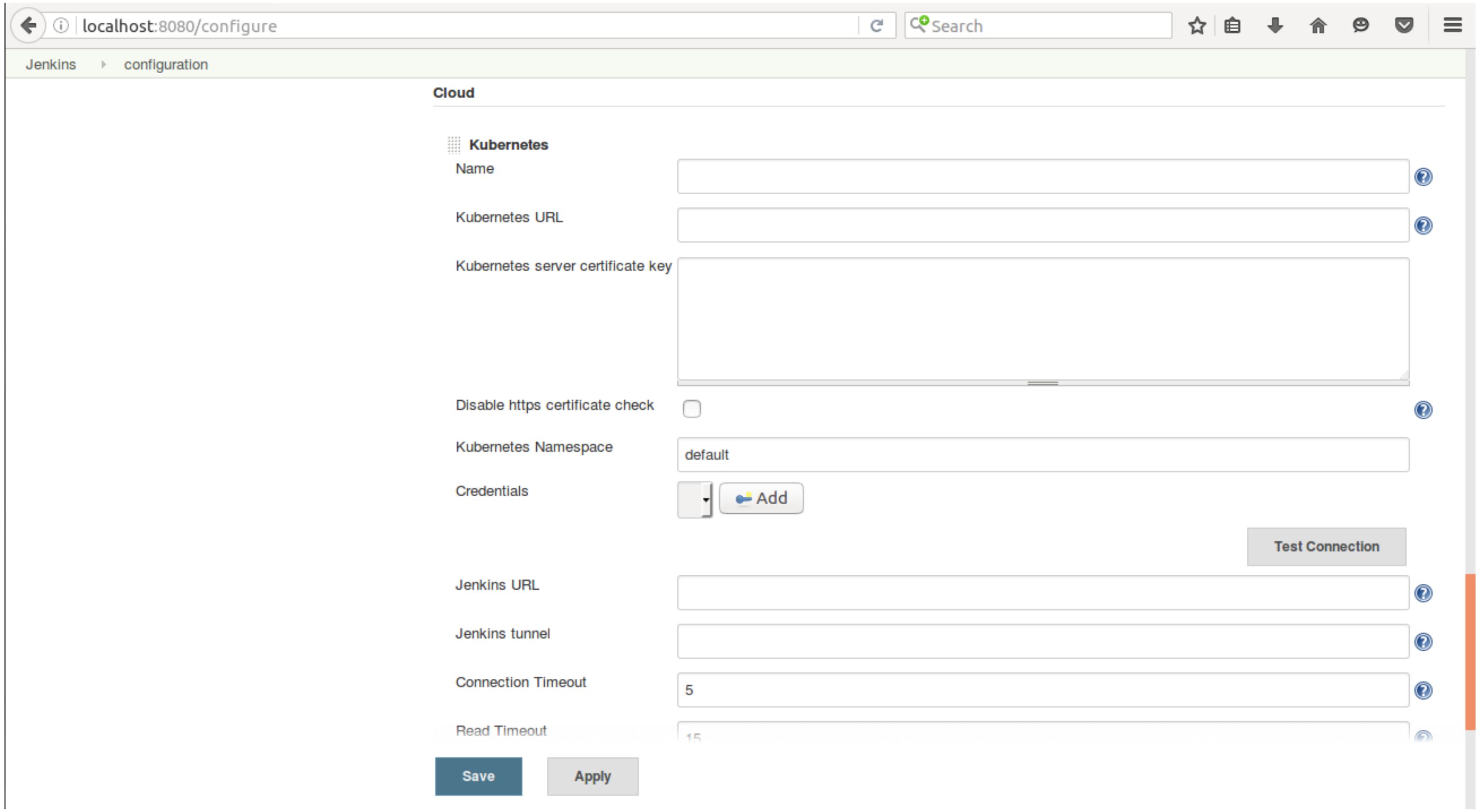
Page generated: Jul 29, 2016 9:46:44 AM REST API Jenkins ver. 1.642.2

Behind the scenes

- Jenkins image is automatically connected to the Jenkins master
- Slaves are launched using JNLP
- Some of the environment variables that are automatically injected:
- JENKINS_URL: Jenkins web interface url
- JENKINS_JNLP_URL: url for the jnlp definition of the specific slave
- JENKINS_SECRET: the secret key for authentication

How to configure it?

- Go to Manage Jenkins -> Configure System -> Cloud -> Kubernetes



localhost:8080/configure

Jenkins › configuration

Cloud

Kubernetes

Name

Kubernetes URL

Kubernetes server certificate key

Disable https certificate check □

Kubernetes Namespace: default

Credentials

Jenkins URL

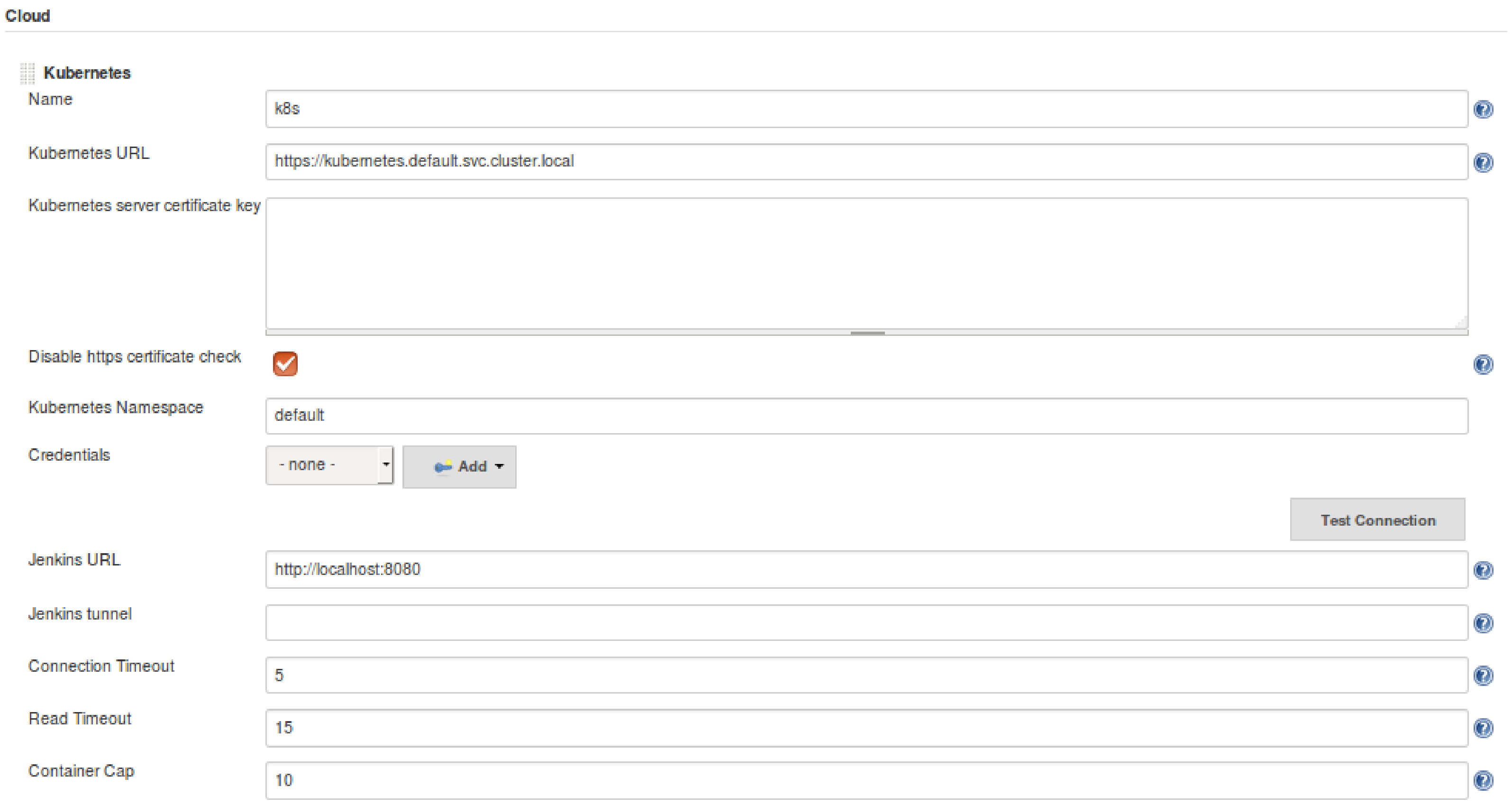
Jenkins tunnel

Connection Timeout: 5

Read Timeout: 15

How to configure it?

- Sample Configuration file



Cloud

Kubernetes

Name: k8s

Kubernetes URL: https://kubernetes.default.svc.cluster.local

Kubernetes server certificate key:

Disable https certificate check: ☑

Kubernetes Namespace: default

Credentials: - none -

Jenkins URL: http://localhost:8080

Jenkins tunnel:

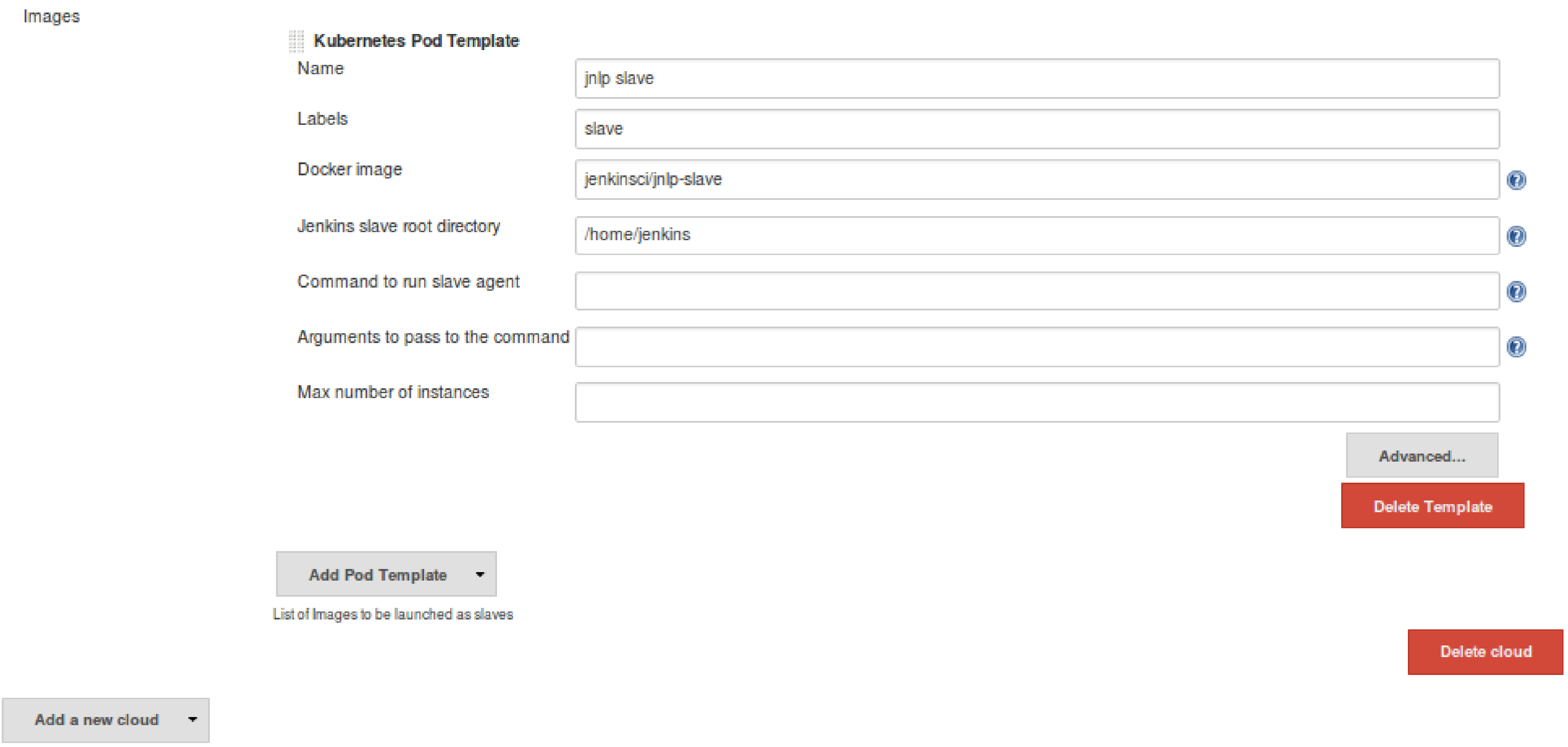
Connection Timeout: 5

Read Timeout: 15

Container Cap: 10

How to configure it?

- Sample Configuration file for slave pod template



Images

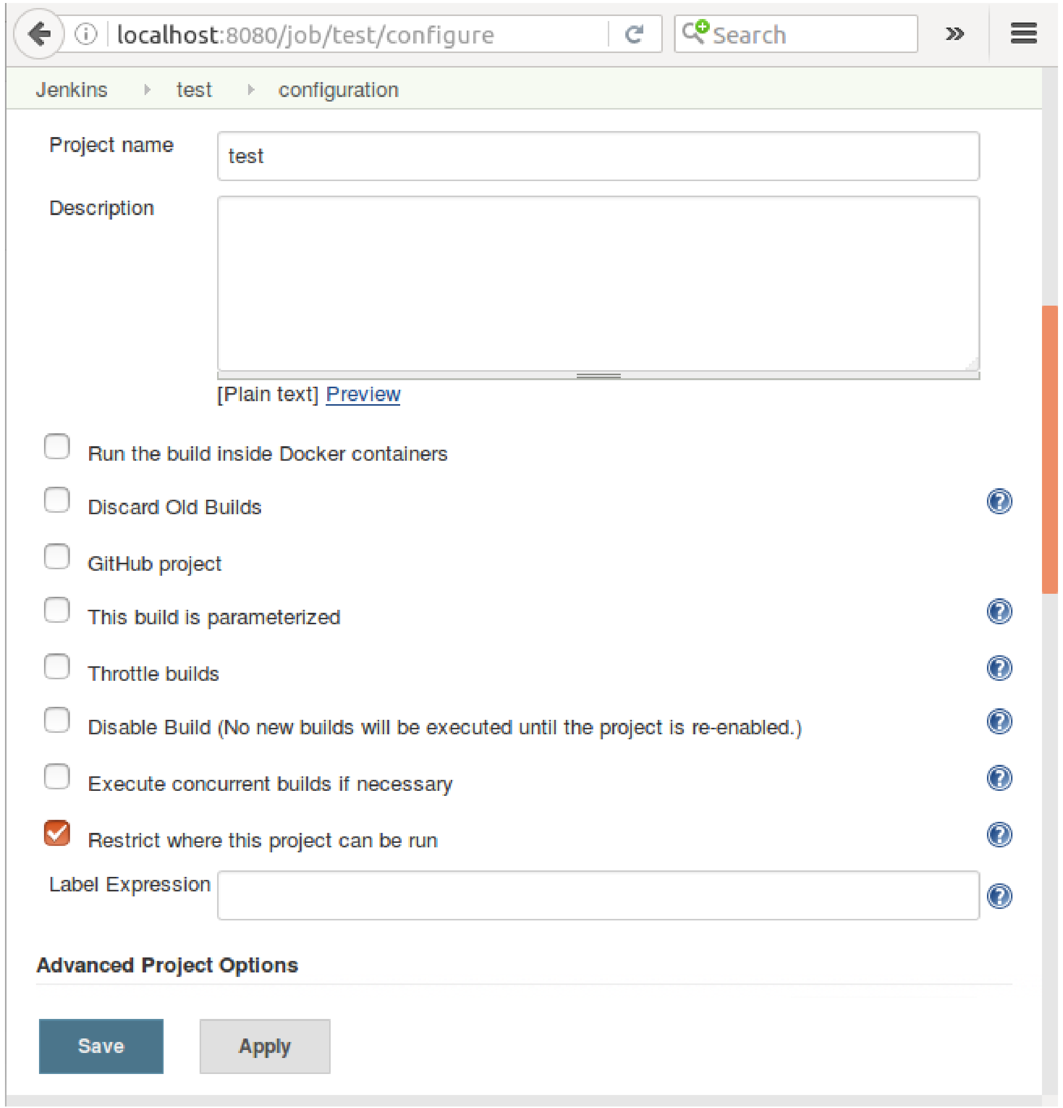
Kubernetes Pod Template

Name	jnlp slave
Labels	slave
Docker image	jenkinsci/jnlp-slave
Jenkins slave root directory	/home/jenkins
Command to run slave agent	
Arguments to pass to the command	
Max number of instances	

List of Images to be launched as slaves

How to configure it with adding new jobs ?

- Sample configuration while adding a new job



Project name: test

Description:

[Plain text] Preview

□ Run the build inside Docker containers

□ Discard Old Builds

□ GitHub project

□ This build is parameterized

□ Throttle builds

□ Disable Build (No new builds will be executed until the project is re-enabled.)

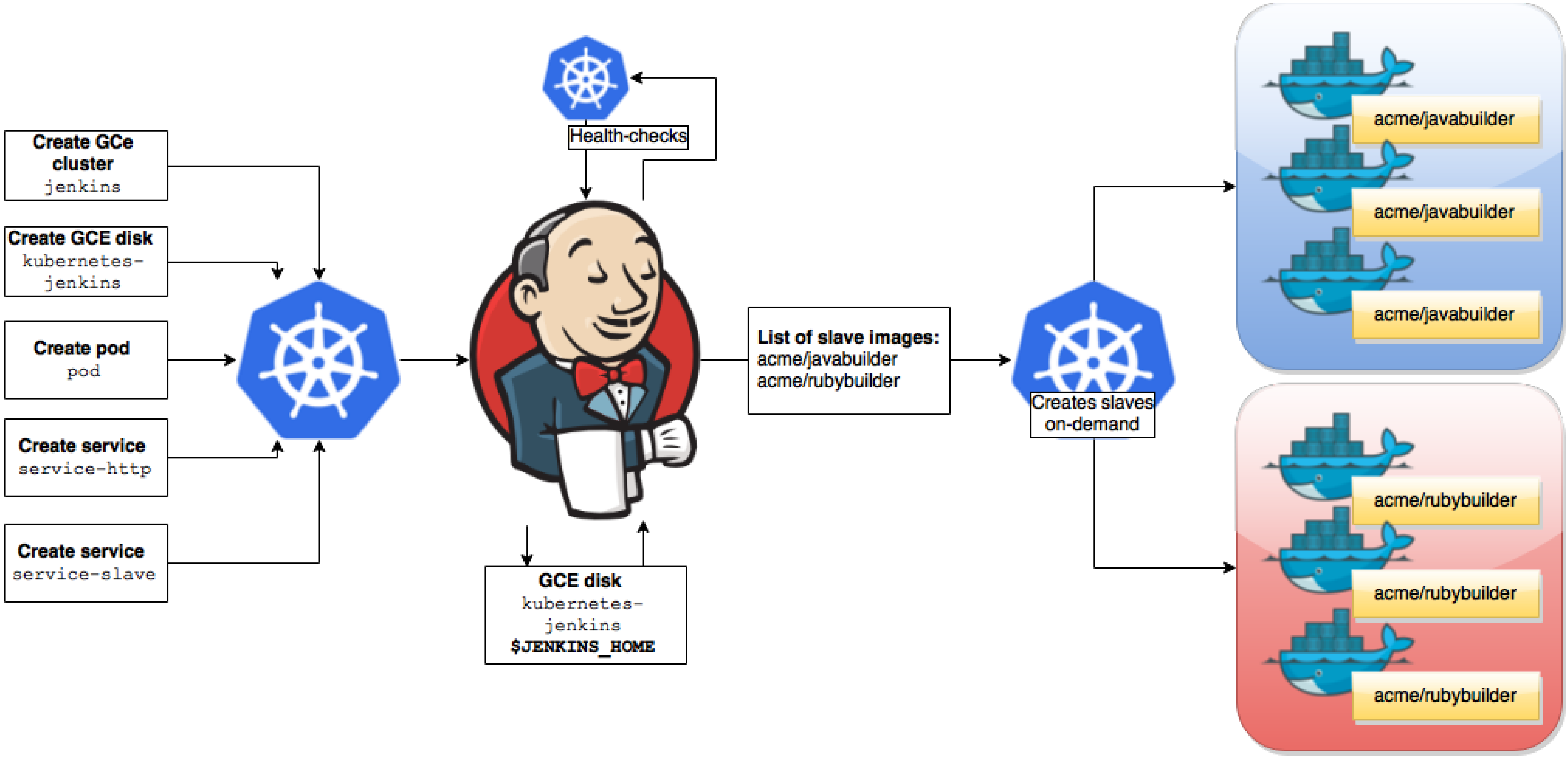
□ Execute concurrent builds if necessary

☑ Restrict where this project can be run

Label Expression:

Advanced Project Options

Continuous Delivery with Kubernetes



Create GCe cluster
`jenkins`

Create GCE disk
`kubernetes-jenkins`

Create pod
`pod`

Create service
`service-http`

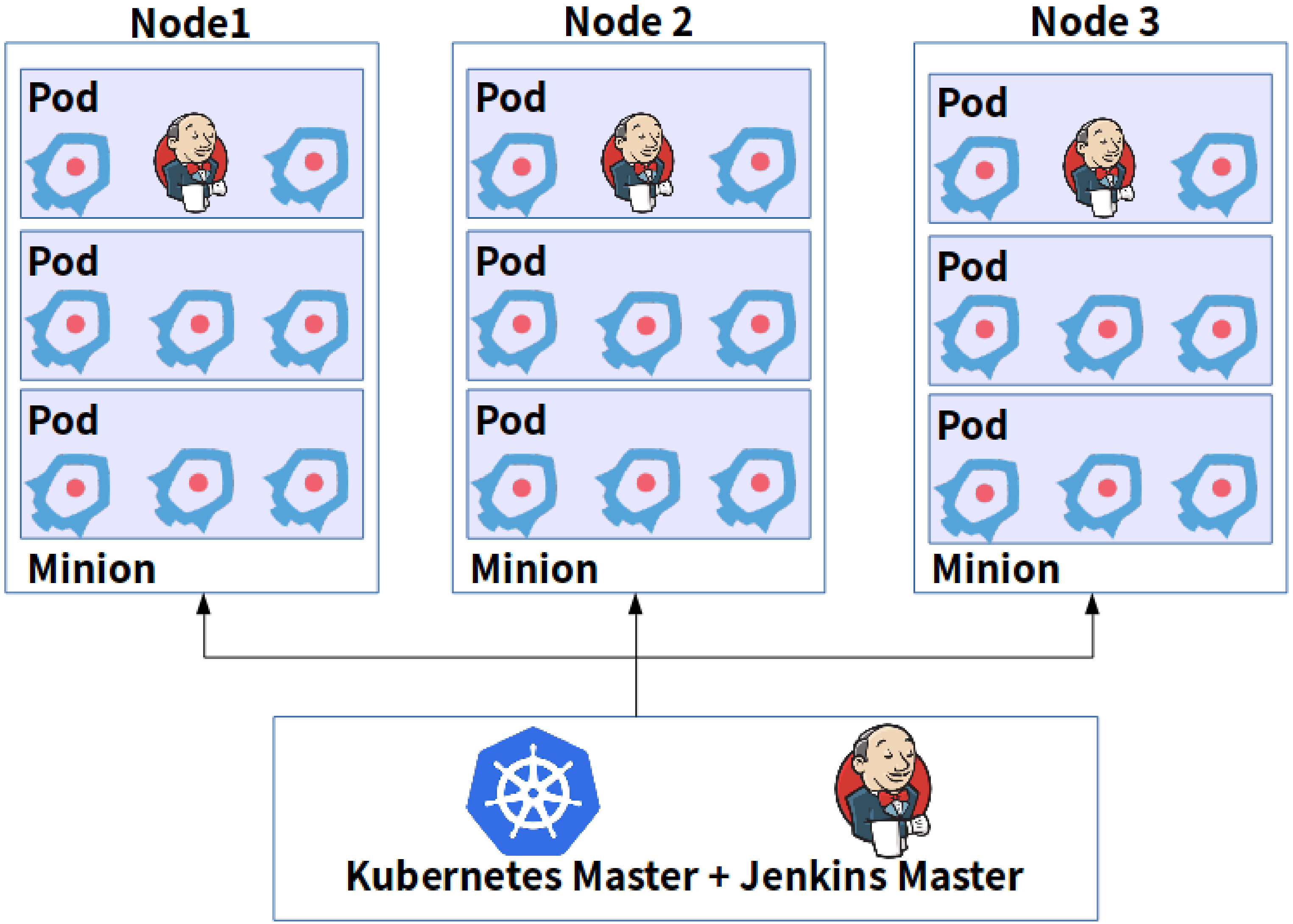
Create service
`service-slave`

List of slave images:
acme/javabuilder
acme/rubybuilder

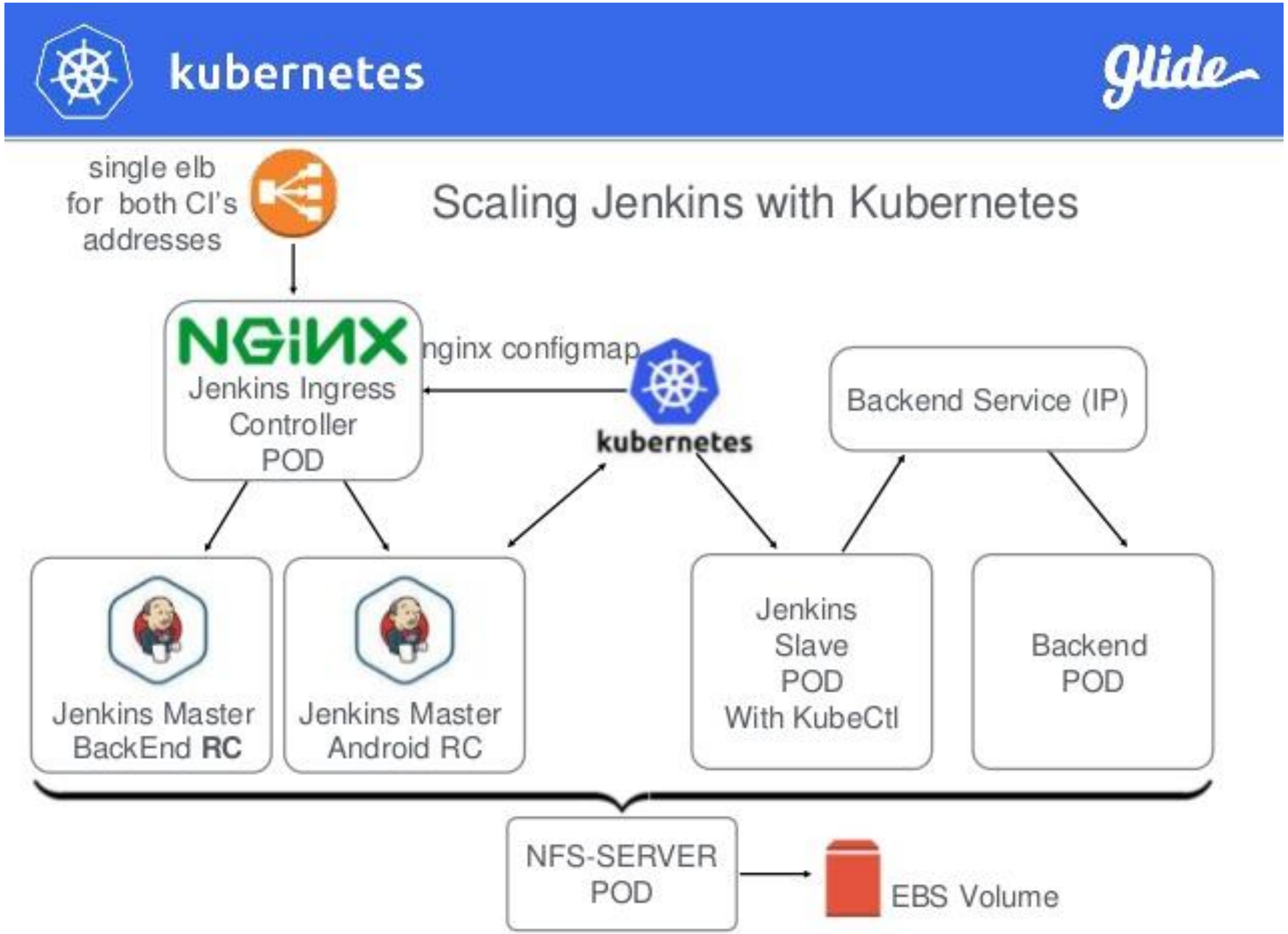
Rkt by CoreOS

- Container project by CoreOS
- Supports ACI and pods by default
- Main focus is to build containers keeping in mind the security aspect for containers.

Workflow with rkt a.k.a. rktnetes



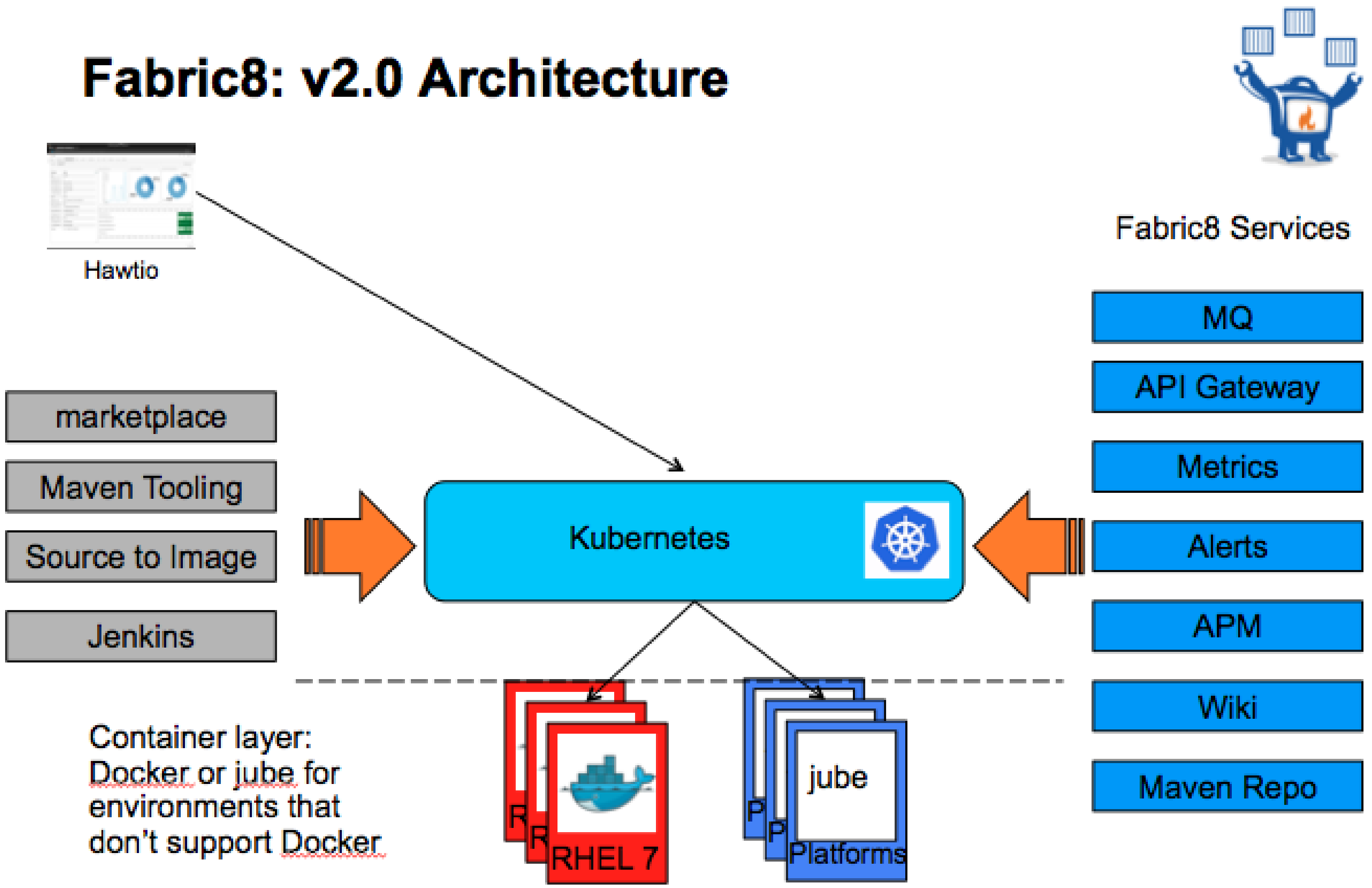
Practical example with Kubernetes


kubernetes
Glide
single elb for both CI's addresses
Scaling Jenkins with Kubernetes
NGINX
Jenkins Ingress Controller POD
nginx configmap
kubernetes
Backend Service (IP)
Jenkins Master BackEnd RC
Jenkins Master Android RC
Jenkins Slave POD With KubeCtl
Backend POD
NFS-SERVER POD
EBS Volume

Fabric8

- Microservices platform by RedHat Guys
- Uses openshift, Jenkins and kubernetes
- Contains several different apps for integrations.

Fabric8 workflow

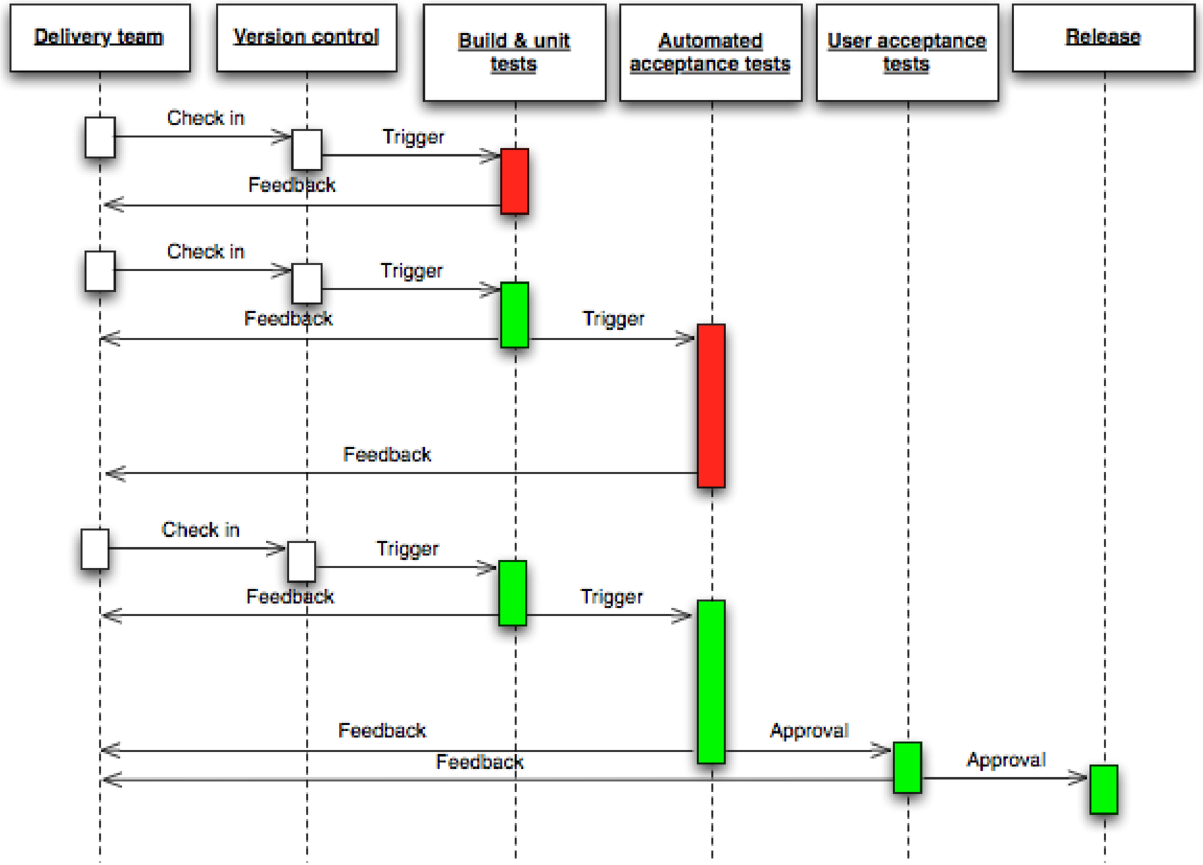


Fabric8: v2.0 Architecture

Container layer:
Docker or jube for
environments that
don't support Docker

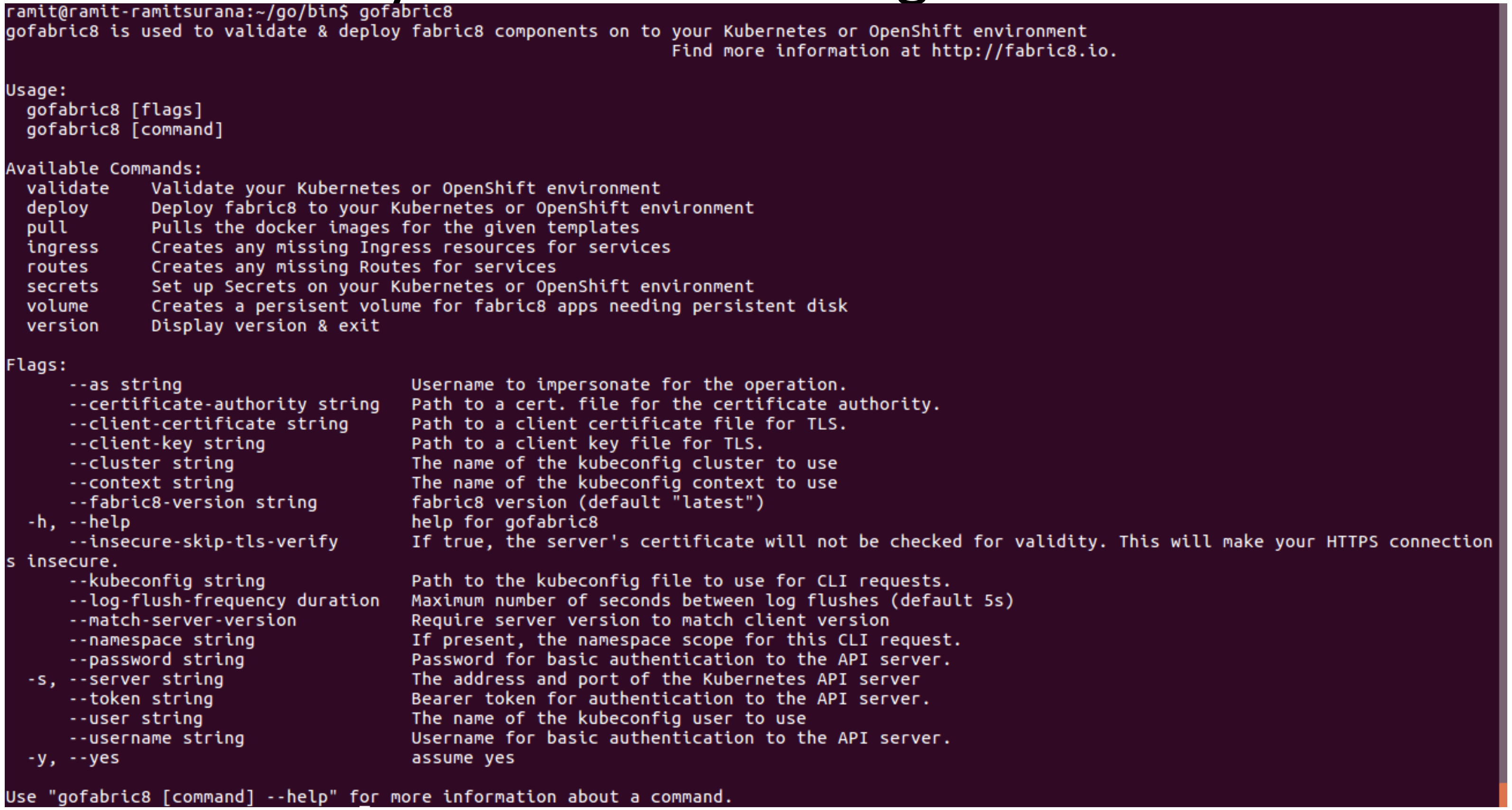
Features

- CI/CD part of fabric8:
- Jenkins
- Gogs
- Fabric8 registry
- Nexus
- SonarQube



How to do it ?

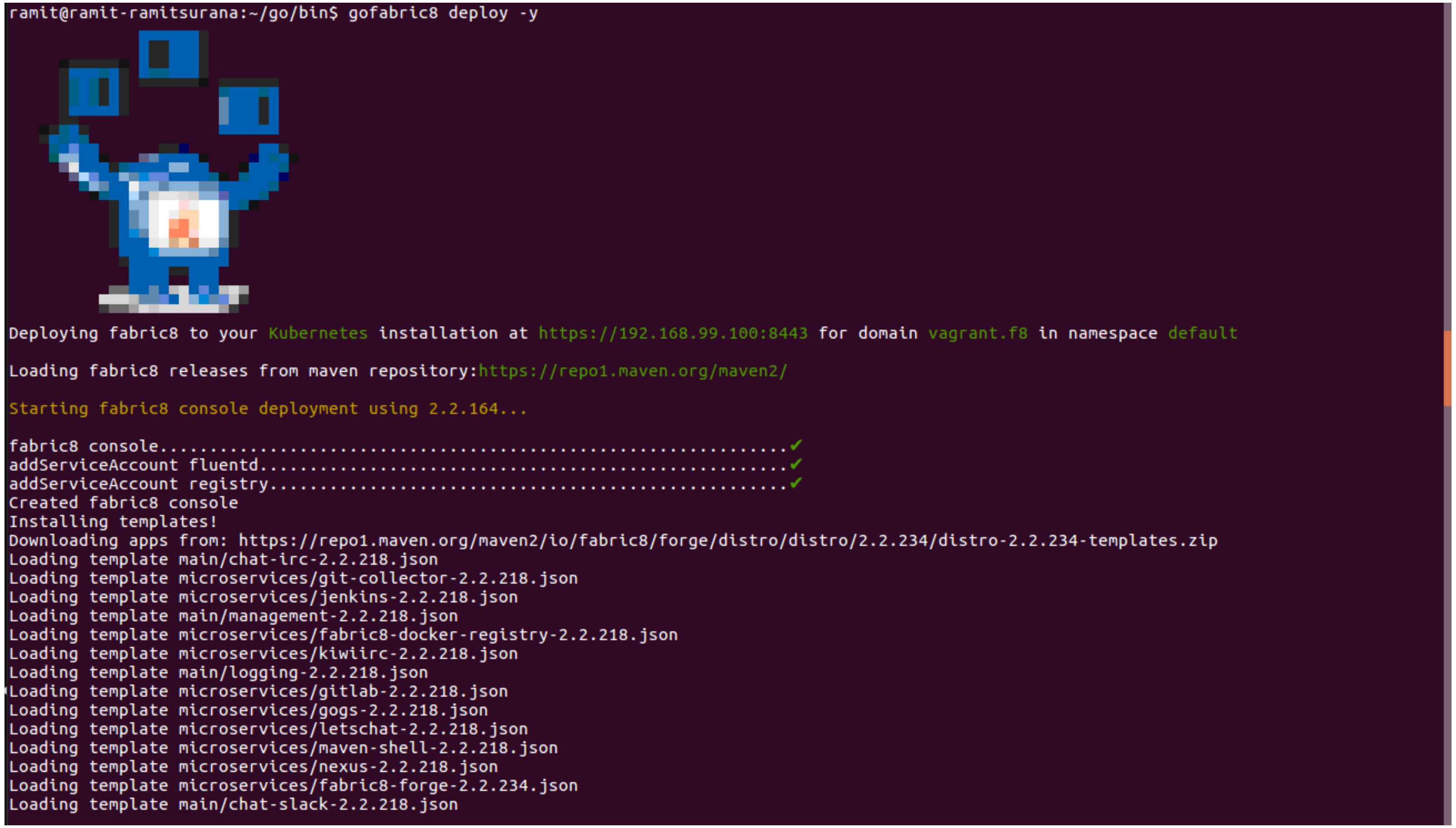
- Install gofabric8 on your local $PATH
- Ensure that you have a running kubernetes cluster



```
ramit@ramit-ramitsurana:~/go/bin$ gofabric8
gofabric8 is used to validate & deploy fabric8 components on to your Kubernetes or OpenShift environment
                                                                Find more information at http://fabric8.io.

Usage:
  gofabric8 [flags]
  gofabric8 [command]

Available Commands:
  validate    Validate your Kubernetes or OpenShift environment
  deploy      Deploy fabric8 to your Kubernetes or OpenShift environment
  pull        Pulls the docker images for the given templates
  ingress     Creates any missing Ingress resources for services
  routes      Creates any missing Routes for services
  secrets     Set up Secrets on your Kubernetes or OpenShift environment
  volume      Creates a persisent volume for fabric8 apps needing persistent disk
  version     Display version & exit

Flags:
      --as string                        Username to impersonate for the operation.
      --certificate-authority string     Path to a cert. file for the certificate authority.
      --client-certificate string        Path to a client certificate file for TLS.
      --client-key string                Path to a client key file for TLS.
      --cluster string                   The name of the kubeconfig cluster to use
      --context string                   The name of the kubeconfig context to use
      --fabric8-version string           fabric8 version (default "latest")
  -h, --help                             help for gofabric8
      --insecure-skip-tls-verify         If true, the server's certificate will not be checked for validity. This will make your HTTPS connection
s insecure.
      --kubeconfig string                Path to the kubeconfig file to use for CLI requests.
      --log-flush-frequency duration     Maximum number of seconds between log flushes (default 5s)
      --match-server-version             Require server version to match client version
      --namespace string                 If present, the namespace scope for this CLI request.
      --password string                  Password for basic authentication to the API server.
  -s, --server string                    The address and port of the Kubernetes API server
      --token string                     Bearer token for authentication to the API server.
      --user string                      The name of the kubeconfig user to use
      --username string                  Username for basic authentication to the API server.
  -y, --yes                              assume yes

Use "gofabric8 [command] --help" for more information about a command.
```

How to do it ?

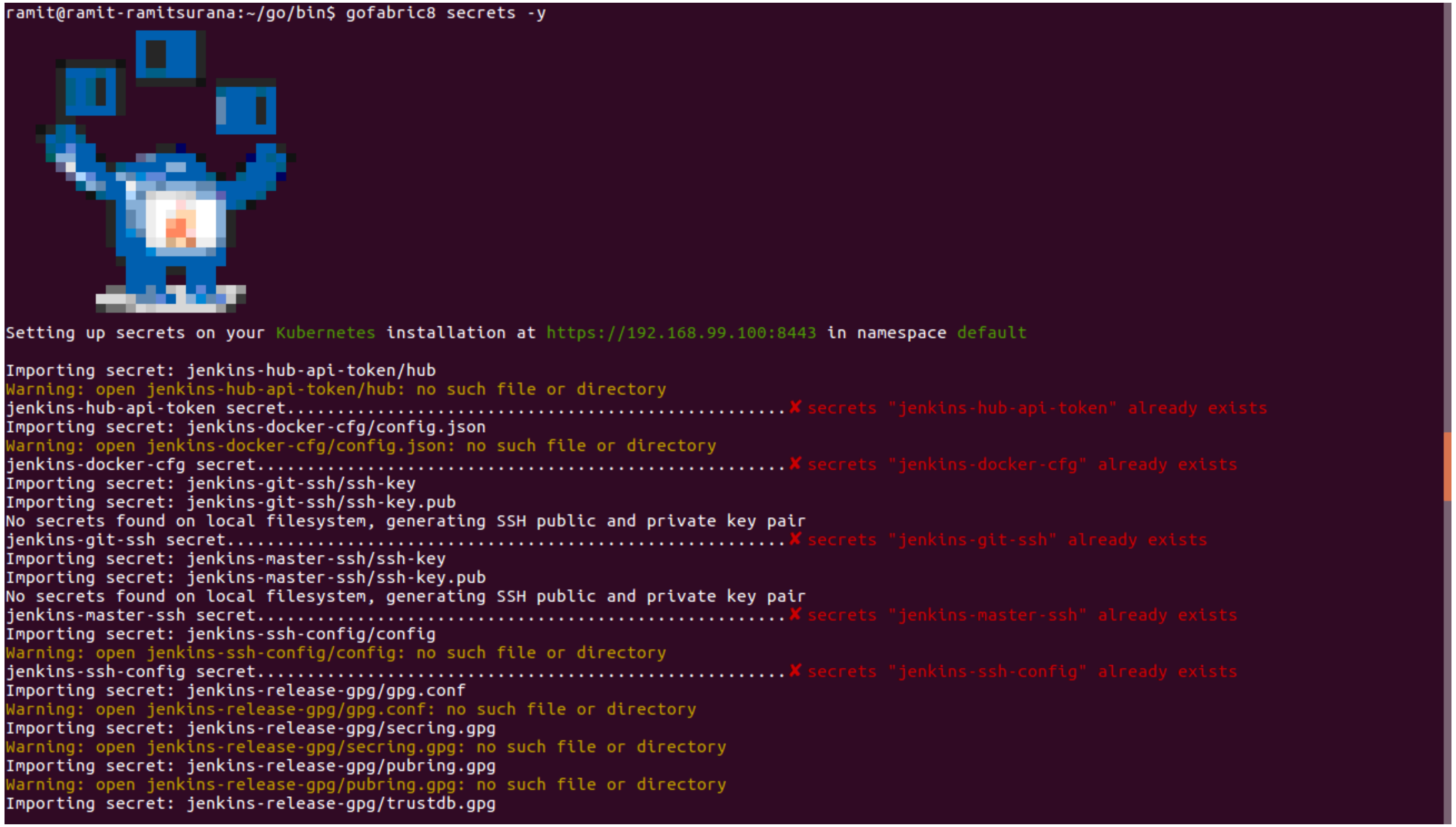
- Run $ gofabric8 deploy –y



```
ramit@ramit-ramitsurana:~/go/bin$ gofabric8 deploy -y

Deploying fabric8 to your Kubernetes installation at https://192.168.99.100:8443 for domain vagrant.f8 in namespace default

Loading fabric8 releases from maven repository:https://repo1.maven.org/maven2/

Starting fabric8 console deployment using 2.2.164...

fabric8 console..............................................................✔
addServiceAccount fluentd....................................................✔
addServiceAccount registry...................................................✔
Created fabric8 console
Installing templates!
Downloading apps from: https://repo1.maven.org/maven2/io/fabric8/forge/distro/distro/2.2.234/distro-2.2.234-templates.zip
Loading template main/chat-irc-2.2.218.json
Loading template microservices/git-collector-2.2.218.json
Loading template microservices/jenkins-2.2.218.json
Loading template main/management-2.2.218.json
Loading template microservices/fabric8-docker-registry-2.2.218.json
Loading template microservices/kiwiirc-2.2.218.json
Loading template main/logging-2.2.218.json
Loading template microservices/gitlab-2.2.218.json
Loading template microservices/gogs-2.2.218.json
Loading template microservices/letschat-2.2.218.json
Loading template microservices/maven-shell-2.2.218.json
Loading template microservices/nexus-2.2.218.json
Loading template microservices/fabric8-forge-2.2.234.json
Loading template main/chat-slack-2.2.218.json
```

How to do it ?

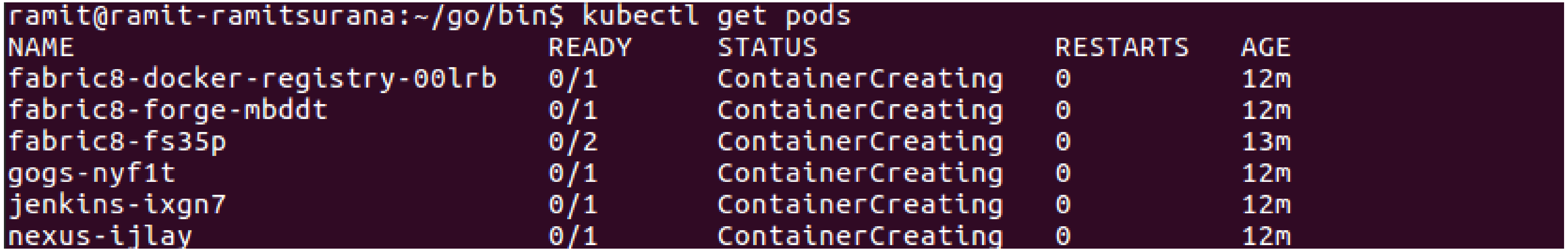
- Run $ gofabric8 secrets –y



```
ramit@ramit-ramitsurana:~/go/bin$ gofabric8 secrets -y

Setting up secrets on your Kubernetes installation at https://192.168.99.100:8443 in namespace default

Importing secret: jenkins-hub-api-token/hub
Warning: open jenkins-hub-api-token/hub: no such file or directory
jenkins-hub-api-token secret..................................................✗ secrets "jenkins-hub-api-token" already exists
Importing secret: jenkins-docker-cfg/config.json
Warning: open jenkins-docker-cfg/config.json: no such file or directory
jenkins-docker-cfg secret.....................................................✗ secrets "jenkins-docker-cfg" already exists
Importing secret: jenkins-git-ssh/ssh-key
Importing secret: jenkins-git-ssh/ssh-key.pub
No secrets found on local filesystem, generating SSH public and private key pair
jenkins-git-ssh secret........................................................✗ secrets "jenkins-git-ssh" already exists
Importing secret: jenkins-master-ssh/ssh-key
Importing secret: jenkins-master-ssh/ssh-key.pub
No secrets found on local filesystem, generating SSH public and private key pair
jenkins-master-ssh secret.....................................................✗ secrets "jenkins-master-ssh" already exists
Importing secret: jenkins-ssh-config/config
Warning: open jenkins-ssh-config/config: no such file or directory
jenkins-ssh-config secret.....................................................✗ secrets "jenkins-ssh-config" already exists
Importing secret: jenkins-release-gpg/gpg.conf
Warning: open jenkins-release-gpg/gpg.conf: no such file or directory
Importing secret: jenkins-release-gpg/secring.gpg
Warning: open jenkins-release-gpg/secring.gpg: no such file or directory
Importing secret: jenkins-release-gpg/pubring.gpg
Warning: open jenkins-release-gpg/pubring.gpg: no such file or directory
Importing secret: jenkins-release-gpg/trustdb.gpg
```

How to do it ?

- $ Kubectl get pods



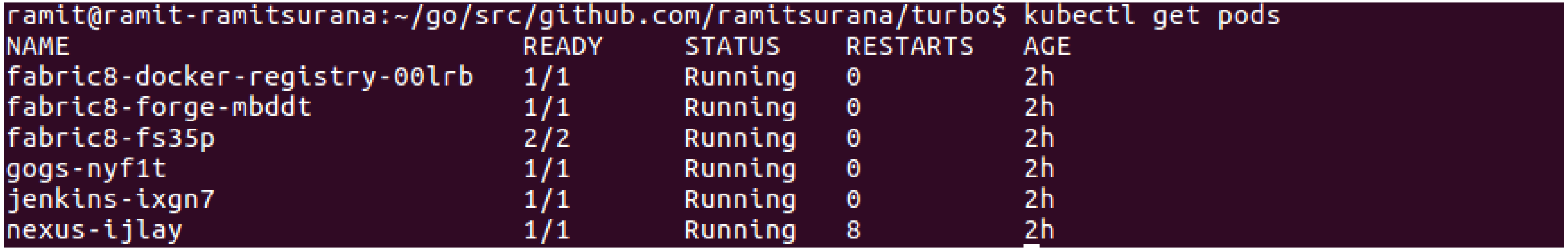
```
ramit@ramit-ramitsurana:~/go/bin$ kubectl get pods
NAME                              READY     STATUS              RESTARTS   AGE
fabric8-docker-registry-00lrb     0/1       ContainerCreating   0          12m
fabric8-forge-mbddt               0/1       ContainerCreating   0          12m
fabric8-fs35p                     0/2       ContainerCreating   0          13m
gogs-nyf1t                        0/1       ContainerCreating   0          12m
jenkins-ixgn7                     0/1       ContainerCreating   0          12m
nexus-ijlay                       0/1       ContainerCreating   0          12m
```

This will take a while.So try going out for a Coffee :)

- In case anything fails try using $ kubectl describe pods.

How to do it ?

- If everything works out fine, then you should see

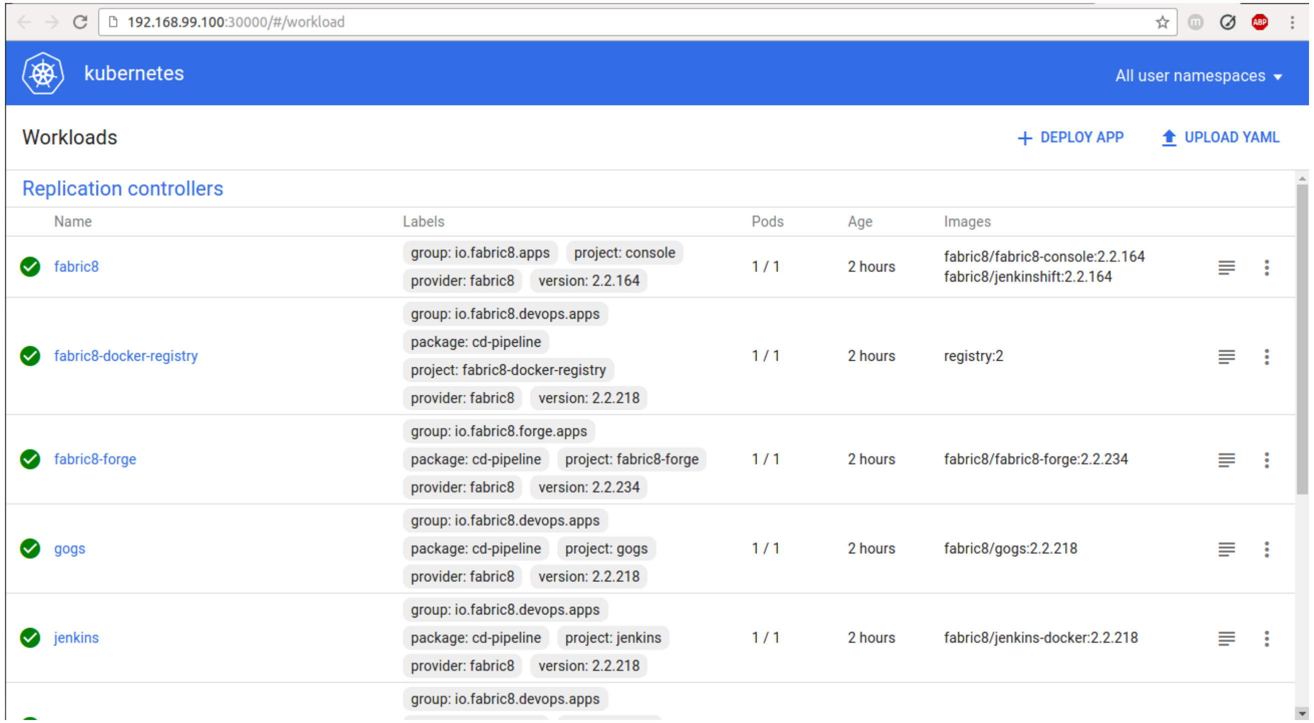


```
ramit@ramit-ramitsurana:~/go/src/github.com/ramitsurana/turbo$ kubectl get pods
NAME                                READY     STATUS    RESTARTS   AGE
fabric8-docker-registry-00lrb       1/1       Running   0          2h
fabric8-forge-mbddt                 1/1       Running   0          2h
fabric8-fs35p                       2/2       Running   0          2h
gogs-nyf1t                          1/1       Running   0          2h
jenkins-ixgn7                       1/1       Running   0          2h
nexus-ijlay                         1/1       Running   8          2h
```

- Try using Kubernetes dashboard for the same and fabric8 console to see your results.

How to do it ?

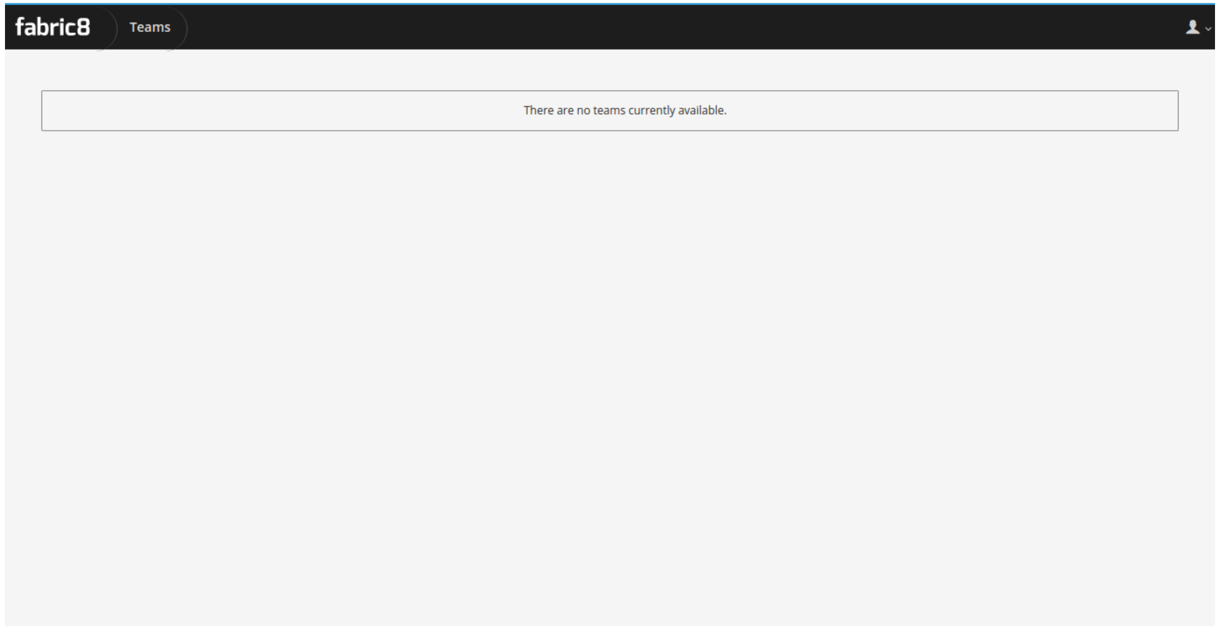
- Kubernetes Dashboard
- Fabric8 console



Workloads

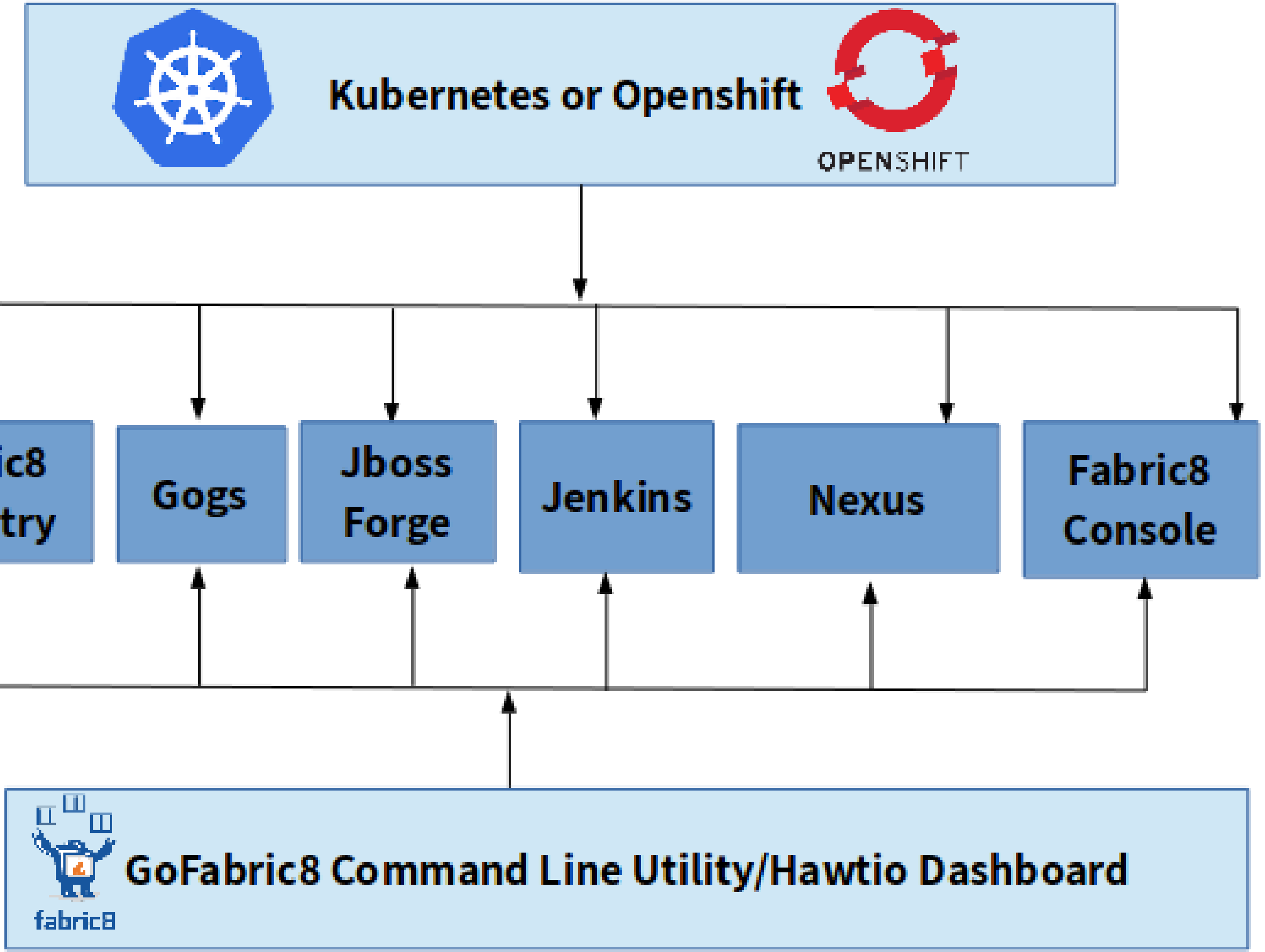
Replication controllers

Name	Labels	Pods	Age	Images
fabric8	group: io.fabric8.apps project: console provider: fabric8 version: 2.2.164	1 / 1	2 hours	fabric8/fabric8-console:2.2.164 fabric8/jenkinshift:2.2.164
fabric8-docker-registry	group: io.fabric8.devops.apps package: cd-pipeline project: fabric8-docker-registry provider: fabric8 version: 2.2.218	1 / 1	2 hours	registry:2
fabric8-forge	group: io.fabric8.forge.apps package: cd-pipeline project: fabric8-forge provider: fabric8 version: 2.2.234	1 / 1	2 hours	fabric8/fabric8-forge:2.2.234
gogs	group: io.fabric8.devops.apps package: cd-pipeline project: gogs provider: fabric8 version: 2.2.218	1 / 1	2 hours	fabric8/gogs:2.2.218
jenkins	group: io.fabric8.devops.apps package: cd-pipeline project: jenkins provider: fabric8 version: 2.2.218	1 / 1	2 hours	fabric8/jenkins-docker:2.2.218
	group: io.fabric8.devops.apps			

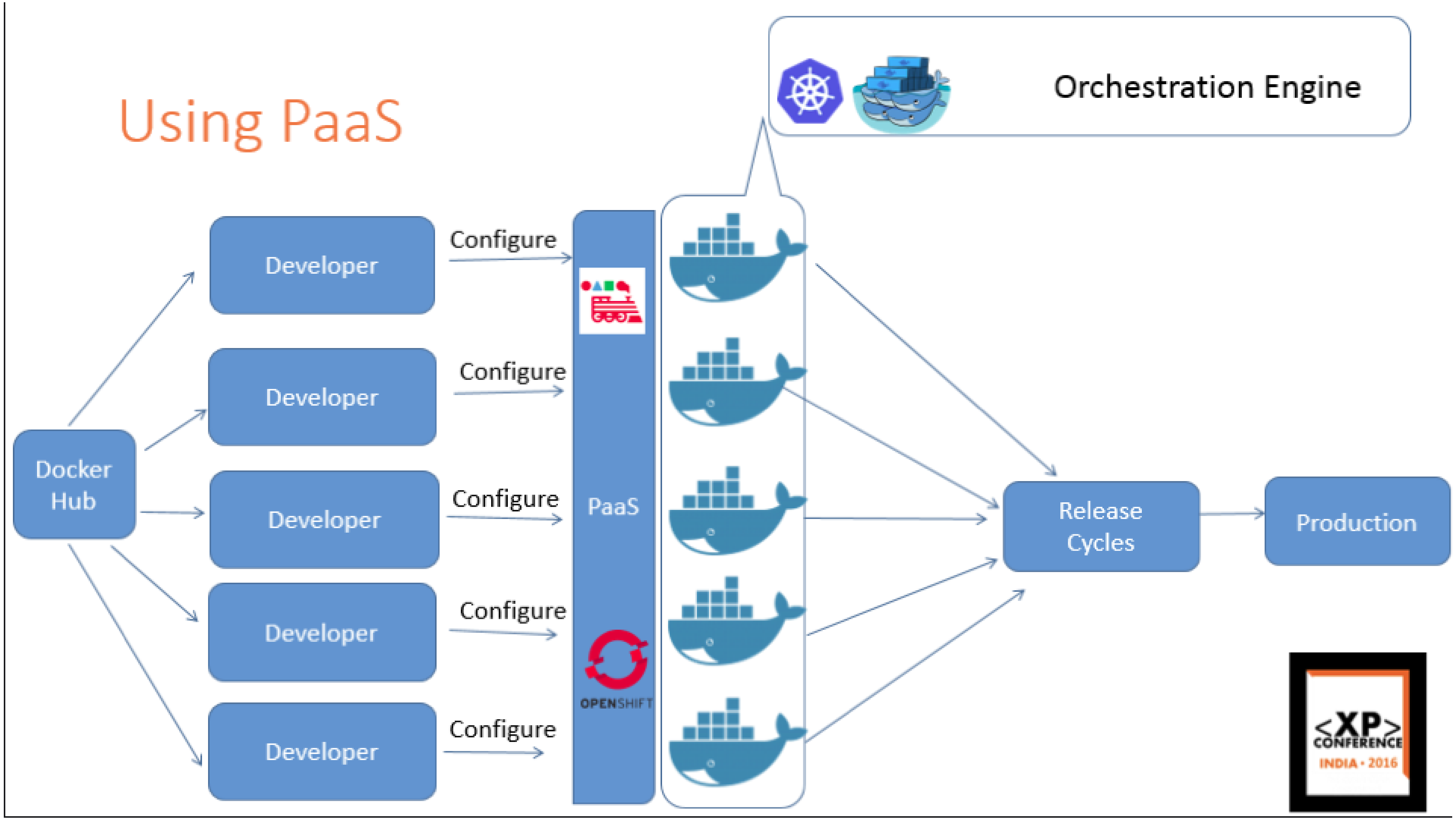


There are no teams currently available.

Behind deploying Fabric8



The Final Picture



Other Tools

- TeamCity
- Codeship
- Travis CI
- Circle CI
- Drone.io
- Semaphore

and many more

Awesome-Kubernetes

- Official list for the Kubernetes Community
- Awesome collection of resources of kubernetes.
- Find articles, blogs, videos, conferences and much more.
- Find out more at:

http://github.com/ramitsurana/awesome-kubernetes

Awesome-Kubernetes

awesome | build passing | License CC 4.0

A curated list for awesome kubernetes sources Inspired by @sindresorhus' awesome

Future of Kubernetes

- Helm
- Kubernetes Chart
- Minikube
- Kops
- Kubedash
- Persistent Storage
- Use of Stateful Containers

And many more...

Got any Questions ?

Thank You

Happy Birthday
Kubernetes !

May you live a thousand
years old !