

Task 1. Installing Jenkins server

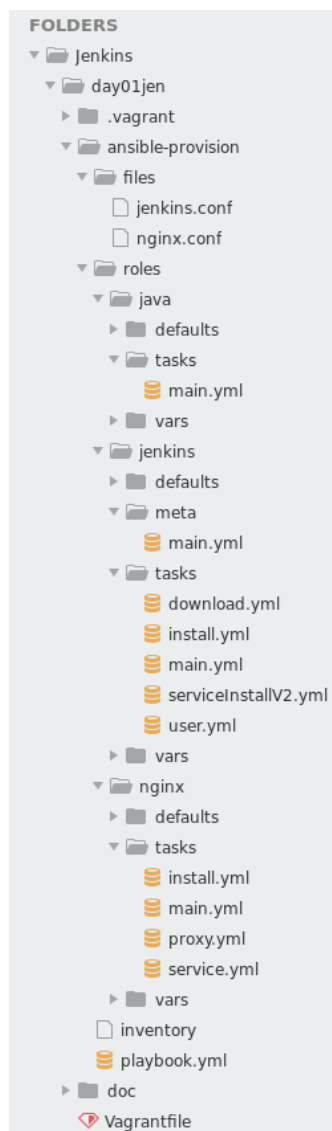
Review

Develop Vagrantfile to spin up Linux box with provisioned via ansible Jenkins server and perform post-installation configuration. Perform basic security configuration.

Task

1. Ansible roles should do:

- Install prerequisites for Jenkins server (Java) – (separate ansible role)
- Install Jenkins (separate ansible role):
 - a. Jenkins installed to /opt/jenkins folder



Picture 1.1

b. Jenkins is running from jenkins user (options for tuning of Jenkins can be taken from slides)

```
-rw-r--r-- 1 jenkins jenkins 1640 Jul 26 15:05 config.xml
-rw-r--r-- 1 jenkins jenkins 156 Jul 26 15:05 hudson.model.UpdateCenter.xml
-rw----- 1 jenkins jenkins 1712 Jul 26 15:05 identity.key.enc
-rw-r--r-- 1 jenkins jenkins 7 Jul 26 15:05 jenkins.install.UpgradeWizard.state
-rw-r--r-- 1 jenkins jenkins 171 Jul 26 15:05 jenkins.telemetry.Correlator.xml
drwxr-xr-x 2 jenkins jenkins 6 Jul 26 15:05 jobs
-rw-r--r-- 1 jenkins jenkins 0 Jul 26 15:05 .lastStarted
drwxr-xr-x 3 jenkins jenkins 19 Jul 26 15:05 logs
-rw-r--r-- 1 jenkins jenkins 907 Jul 26 15:05 nodeMonitors.xml
drwxr-xr-x 2 jenkins jenkins 6 Jul 26 15:05 nodes
drwxr-xr-x 2 jenkins jenkins 6 Jul 26 15:05 plugins
-rw-r--r-- 1 jenkins jenkins 64 Jul 26 15:05 secret.key
-rw-r--r-- 1 jenkins jenkins 0 Jul 26 15:05 secret.key.not-so-secret
drwx----- 4 jenkins jenkins 265 Jul 26 15:05 secrets
drwxr-xr-x 2 jenkins jenkins 67 Jul 26 15:05 updates
drwxr-xr-x 2 jenkins jenkins 24 Jul 26 15:05 userContent
drwxr-xr-x 3 jenkins jenkins 56 Jul 26 15:05 users
[vagrant@centos-jenkins ~]$
```

Picture 1.2

- Install and configure Nginx server for navigation to Jenkins without port in URL. URL should have the following pattern – http://jenkins (separate ansible role)

```
student@EPBYMINW0910:~/devosplab/Jenkins/day01jen/ansible-provision
student@EPBYMINW0910:~/devosplab/Jenkins/day01jen/ansible-provision 130x27
skipping: [jenkins]

TASK [jenkins : Add Jenkins user] *****
skipping: [jenkins]

TASK [jenkins : Start Jenkins Service] *****
ok: [jenkins]

TASK [nginx : Add Epel-Release Repo] *****
ok: [jenkins]

TASK [nginx : Install NGINX Manually] *****
ok: [jenkins]

TASK [nginx : Configurate NGINX To Work With IPv4] *****
ok: [jenkins]

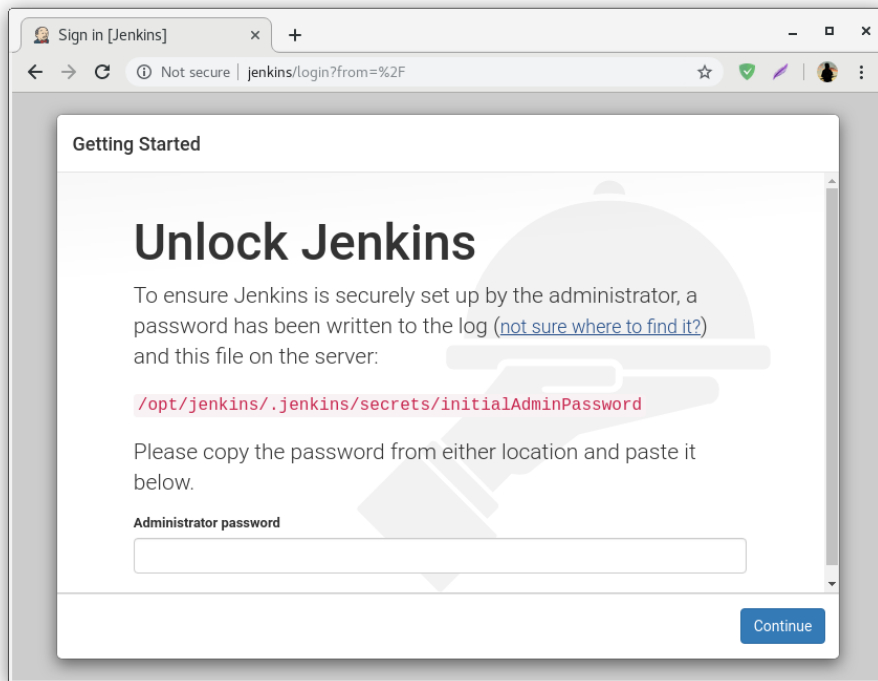
TASK [nginx : Set Proxy Path] *****
ok: [jenkins]

TASK [nginx : Start NGINX Service] *****
ok: [jenkins]

PLAY RECAP *****
jenkins      : ok=12  changed=1  unreachable=0  failed=0  skipped=3  rescued=0  ignored=0

[student@EPBYMINW0910 ansible-provision]$
```

Picture 1.3



Picture 1.4 – Jenkins UI

```
student@EPBYMINW0910:~/devosplab/Jenkins/day01jen/ansible-provision
student@EPBYMINW0910:~/devosplab/Jenkins/day01jen/ansible-provision 130x27
[student@EPBYMINW0910 ansible-provision]$ ansible-playbook -i inventory playbook.yml --list-tasks
playbook: playbook.yml

  play #1 (ciservers): ciservers      TAGS: []
    tasks:
      java : Install Java             TAGS: [always, jenkins]
      java : Java Home Settings       TAGS: [always, jenkins]
      jenkins : Download Jenkins RPM package TAGS: [always, jenkins]
      jenkins : Create Jenkins home directory TAGS: [always, jenkins]
      jenkins : Check if rules file exists TAGS: [always, jenkins]
      jenkins : preInstalling Jenkins v2 TAGS: [always, jenkins]
      jenkins : Installing Jenkins v2 TAGS: [always, jenkins]
      jenkins : Add Jenkins user       TAGS: [always, jenkins]
      jenkins : Start Jenkins Service TAGS: [always, jenkins]
      nginx : Add Epel-Release Repo    TAGS: [always, nginx]
      nginx : Install NGINX Manually   TAGS: [always, nginx]
      nginx : Configurate NGINX To Work With IPv4 TAGS: [always, nginx]
      nginx : Set Proxy Path           TAGS: [always, nginx]
      nginx : Start NGINX Service      TAGS: [always, nginx]
[student@EPBYMINW0910 ansible-provision]$ ansible-playbook -i inventory playbook.yml --list-tags
playbook: playbook.yml

  play #1 (ciservers): ciservers      TAGS: []
    TASK TAGS: [always, jenkins, nginx]
[student@EPBYMINW0910 ansible-provision]$
```


Picture 1.5

2. Configure Jenkins settings in global configuration UI (java, authentication, email)

Users




























These users can log into Jenkins. This is a sub set of [this list](#), which also contains auto-created users who really just made some commits on some projects and have no direct Jenkins access.

| User ID | Name |
|--------------------------|----------------------|
| hkanonik | Hleb |



Picture 2

3. Install 6 Jenkins plugins (Green Balls, ChuckNorris, Multiple SCMs plugin, Email Extension Plugin, Role-based Authorization Strategy)

| | |
|--|---|
| PAM Authentication |  Success |
| Oracle Java SE Development Kit Installer |  Success |
| Script Security |  Success |
| Command Agent Launcher |  Success |
| Struts |  Success |
| Pipeline: Step API |  Success |
| SCM API |  Success |
| Pipeline: API |  Success |
| JUnit |  Success |
| bouncycastle API |  Success |
| WMI Windows Agents |  Success |
| Display URL API |  Success |
| Mailer |  Success |
| LDAP |  Success |
| External Monitor Job Type |  Success |
| Matrix Authorization Strategy |  Success |
| OWASP Markup Formatter |  Success |
| Matrix Project |  Success |
| Green Balls |  Success |
| Multiple SCMs |  Success |
| Token Macro |  Success |
| Email Extension |  Success |
| ChuckNorris |  Success |
| Ant |  Success |
| jQuery |  Success |
| Javadoc |  Success |
| jQuery UI |  Success |
| Role-based Authorization Strategy |  Success |

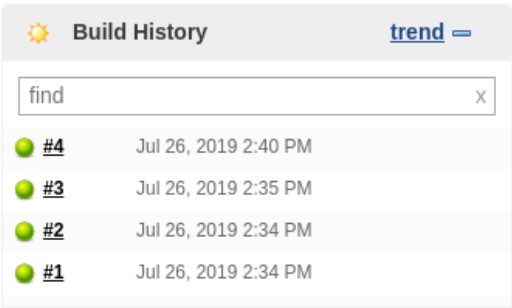
Picture 3








4. Create free-style job with simple build step to output some message to console. Trigger build. Configure job to be build periodically (every 5 min).

Console Output

```
Started by user Hleb
Running as SYSTEM
Building in workspace /opt/jenkins/.jenkins/workspace/free-style-job
[free-style-job] $ /bin/sh -xe /tmp/jenkins3611929069209591094.sh
+ echo 'Hello Jenkins!'
Hello Jenkins!
Finished: SUCCESS
```

Picture 4.1



|  | Build History | trend  |
|---|-----------------------------------|--|
| | <input type="text" value="find"/> |  |
|  | #4 | Jul 26, 2019 2:40 PM |
|  | #3 | Jul 26, 2019 2:35 PM |
|  | #2 | Jul 26, 2019 2:34 PM |
|  | #1 | Jul 26, 2019 2:34 PM |

Picture 4.2

5. Crete 4 free-style jobs: ADMIN-job1, ADMIN-job2, USER-job1, USER-job2

| All  | | | | | | |
|---|---|--------------------------------|-----------------------------------|--------------|---------------|---|
| S | W | Name | Last Success | Last Failure | Last Duration | |
|  |  | ADMIN-job1 | N/A | N/A | N/A |  |
|  |  | ADMIN-job2 | N/A | N/A | N/A |  |
|  |  | free-style-job | 2 min 40 sec - #8 | N/A | 14 ms |  |
|  |  | USER-job1 | N/A | N/A | N/A |  |
|  |  | USER-job2 | 1 min 41 sec - #1 | N/A | 13 ms |  |

Picture 5

6. Setup Jenkins jobs to build of any selected open-source project. Output should be some package (artifact). Project could be even archiving of random files – this will be enough for completion.



Picture 6.1

Console Output

```
Started by user Hleb
Running as SYSTEM
Building in workspace /opt/jenkins/.jenkins/workspace/open-source
[open-source] $ /bin/sh -xe /tmp/jenkins5902672763412900177.sh
+ touch javalogs
+ echo 'java not installed'
+ tar -cvf javalogs-26-07-19.tar ./javalogs
./javalogs
+ ls -la ./
total 16
drwxr-xr-x 2 root root    51 Jul 26 14:28 .
drwxr-xr-x 5 root root    64 Jul 26 14:28 ..
-rw-r--r-- 1 root root    19 Jul 26 14:29 javalogs
-rw-r--r-- 1 root root 10240 Jul 26 14:29 javalogs-26-07-19.tar
Finished: SUCCESS
```

Picture 6.2

7. Configure Role Based Strategy plugin (<https://plugins.jenkins.io/role-strategy>). Guide: <https://github.com/jenkinsci/role-strategy-plugin/blob/master/README.md>.




With 3 users: admin, user, read-only

- admin can do everything.
- user can view, build and configure USER.* jobs only
- read-only can only view USER.* jobs.



Manage Roles



Global roles

| Role | Overall | | Agent | | | | | | | | Job | | | | | | | | Run | | View | | |
|---|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|--|
| | Administer | Read | Build | Configure | Connect | Create | Delete | Disconnect | Provision | Build | Cancel | Configure | Create | Delete | Discover | Read | Workspace | Delete | Update | Configure | Create | Delete | |
|  admin | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | |
|  read-only | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
|  user | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |

Role to add

Add

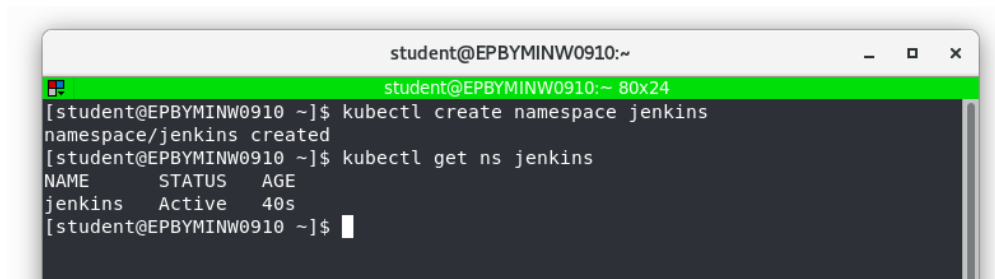
Project roles

| Role | Pattern | Job | | | | | | | | Run | | SCM | |
|---|--------------------------|-------------------------------------|--------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|---|
| | | Build | Cancel | Configure | Create | Delete | Discover | Read | Workspace | Delete | Update | Tag | |
|  read-only | "USER.*" | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |  |
|  user | "USER.*" | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |  |

Picture 6

Task 2. Installing Jenkins server to Kubernetes (Optional for now)

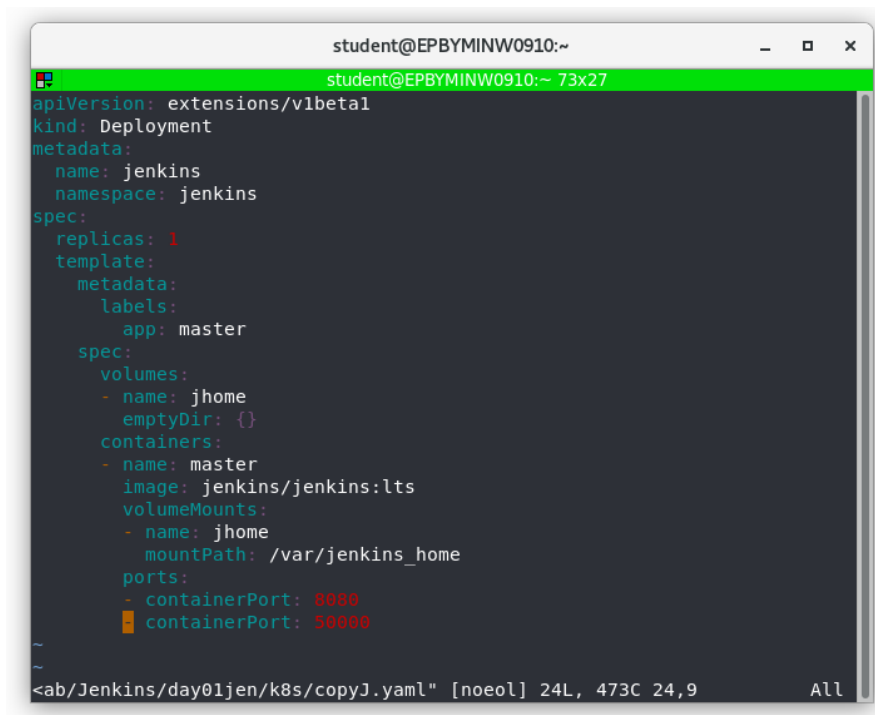
1) Create separate namespace: jenkins

A terminal window titled 'student@EPBYMINW0910:~' with a green header bar. The terminal shows the command 'kubectl create namespace jenkins' being executed, followed by 'namespace/jenkins created'. Then, 'kubectl get ns jenkins' is executed, showing a table with columns NAME, STATUS, and AGE. The row for 'jenkins' shows 'Active' status and '40s' age. The prompt returns to the user.

```
student@EPBYMINW0910:~  
student@EPBYMINW0910:~ 80x24  
[student@EPBYMINW0910 ~]$ kubectl create namespace jenkins  
namespace/jenkins created  
[student@EPBYMINW0910 ~]$ kubectl get ns jenkins  
NAME      STATUS   AGE  
jenkins   Active   40s  
[student@EPBYMINW0910 ~]$
```

Picture 7.1

2) Develop and apply appropriate deployment definition

A terminal window titled 'student@EPBYMINW0910:~' with a green header bar. The terminal displays a YAML configuration for a Jenkins deployment. The configuration includes apiVersion, kind, metadata (name: jenkins, namespace: jenkins), spec (replicas: 1, template with metadata labels app: master, spec with volumes jhome and container master). The master container uses the jenkins/jenkins:lts image, mounts the jhome volume to /var/jenkins_home, and exposes ports 8080 and 50000. The terminal ends with a file path and a status bar.

```
student@EPBYMINW0910:~  
student@EPBYMINW0910:~ 73x27  
apiVersion: extensions/v1beta1  
kind: Deployment  
metadata:  
  name: jenkins  
  namespace: jenkins  
spec:  
  replicas: 1  
  template:  
    metadata:  
      labels:  
        app: master  
    spec:  
      volumes:  
        - name: jhome  
          emptyDir: {}  
      containers:  
        - name: master  
          image: jenkins/jenkins:lts  
          volumeMounts:  
            - name: jhome  
              mountPath: /var/jenkins_home  
          ports:  
            - containerPort: 8080  
            - containerPort: 50000  
~  
<ab/Jenkins/day01jen/k8s/copyJ.yaml" [noeol] 24L, 473C 24,9 All
```

Picture 7.2

3) Develop and apply appropriate service definition (ClusterIP)

```
student@EPBYMINW0910:~  
student@EPBYMINW0910:~ 73x27  
apiVersion: v1  
kind: Service  
metadata:  
  name: jenkins-ui  
  namespace: jenkins  
spec:  
  type: ClusterIP  
  selector:  
    app: master  
  ports:  
    - name: ui  
      port: 80  
      targetPort: 8080
```

<b/Jenkins/day01jen/k8s/service.yml" [noeol] 13L, 186C 1,1 All

Picture 7.3

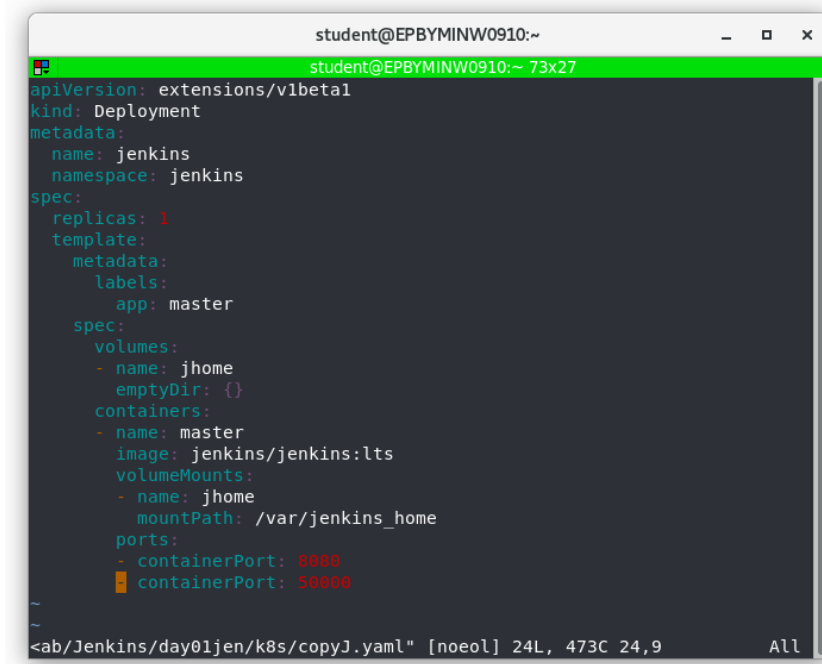
4) Develop and apply ingress rule

```
student@EPBYMINW0910:~  
student@EPBYMINW0910:~ 73x27  
apiVersion: extensions/v1beta1  
kind: Ingress  
metadata:  
  name: jenkins  
  namespace: jenkins  
spec:  
  rules:  
    - host: jenkins.local  
      http:  
        paths:  
          - path: /  
            backend:  
              serviceName: jenkins-ui  
              servicePort: 8686
```

<devosplab/Jenkins/day01jen/k8s/ingress.yml" 14L, 249C 1,1 All

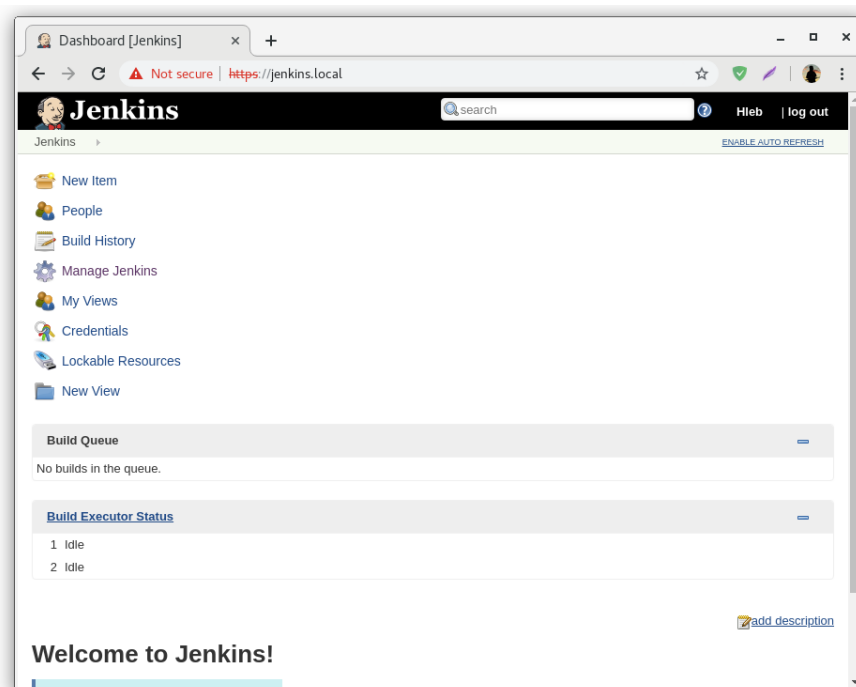
Picture 7.4

5) Develop and apply appropriate pv and pvc definitions (for mount JENKINS_HOME)



```
student@EPBYMINW0910:~  
student@EPBYMINW0910:~ 73x27  
apiVersion: extensions/v1beta1  
kind: Deployment  
metadata:  
  name: jenkins  
  namespace: jenkins  
spec:  
  replicas: 1  
  template:  
    metadata:  
      labels:  
        app: master  
    spec:  
      volumes:  
        - name: jhome  
          emptyDir: {}  
      containers:  
        - name: master  
          image: jenkins/jenkins:ls  
          volumeMounts:  
            - name: jhome  
              mountPath: /var/jenkins_home  
          ports:  
            - containerPort: 8080  
            - containerPort: 50000  
~  
<ab/Jenkins/day01jen/k8s/copyJ.yaml" [noeol] 24L, 473C 24,9 All
```

Picture 7.5



Picture 7.6