Issues observed:

Once we increased our Ram size from 512MB to 1GB, sample job was able to progress but failed at nabu step that is failing to find a file that should be part of Repository

Cloudfees nabu folder is not getting cloned so we manually cloned the reposiotry to get the build succeed.

Created a Docker file

Create a Docker file that will create a custom image with Checkmarx plugin installed in it. Also make modfiyications to create an image which contains jobs in it

For sample, created some config test files in config folder

1. Create the image with custom defined plugins and Jobs
2. use docker build -t "custom-jenkins" .
3. that creats a docker image
4. Make this image available in our Open shift image registry
5. Creat a deployment yaml that makes use of this image .
6. Automate the container deployment through a shell script

Jenkins persistent images is predifined with volume 1 Gb for /var/lib/jenkins i.e, Jenkins home.

There is no option to modify the volume size from webconsole

So I see two options:

1. See if we can update the volume from command line using oc command line tools . We need to redploy jenkins after update to the volume configuration
2. Or create a new mount point with defined sizeand with the help of environment variables using config maps, use the new mount point as Jenkins home.

Commands to list volumes: (Donot perform oc rsh )

1. Logon on to project oc project <project-name>
2. oc get pv (lists of all available persistent volumes configured)
3. To claim a suitable persistent volume and mount it using oc volume dc/nginx --add --claim-size 512M --mount-path /usr/share/nginx/html --name downloads
4. oc get pv will list the new persistent volume claim attached to one of the peristent volumes available
5. oc get pvc to list all persistent volume claims
6. oc volume dc --all to which applications the pvc are attached to.
7. oc volume dc/nginx --remove --name downloads to deattach the volumes . This will not remove the pvc from the list. The command will deattach/unmount the volume from nginx application. So that the conatiner cannot use the mount point
8. oc volume dc --all
9. oc volume dc/nginx --add --type pvc --claim-name pvc-rkly7 --mount-path /usr/share/nginx/html --name downloads remount the same pvc into the applicaiton.
10. oc volume dc/nginx --remove --name downloads and oc delete pvc/pvc-rkly7 to detach and delete the pvc completely from pvc list. If so the complete data will be lost.
11. oc describe pod <podname> to get the pod details on the console
12. oc get pod <podname> -o yaml to get the details in yaml output
13. oc rsh <podname>
14. mount
15. df -h
16. cd /var/lib/jenkins
17. du -h --max-depth=2

**Note: dont run steps 3,7,9,10 . And for the remaining commands please send the output**

1. oc get pvc to list all persistent volume claims in the name space. This will o/p the pvc name, size and mount path.
2. Only jenkins pvc is avilable in the name space and is having 1Gi capacity.
3. *First option:* Tried resizing the pvc using oc volume dc/jenkins --add --overwrite --type pvc --claim-name jenkins -- claim-size 2Gi . But oc is not allwoing it to resize the volume with forbidden error.
4. *Second option:* oc get pvc jenkins -o yaml > pvc.yaml . Get the pvc configuration in yaml format and redirect it to pvc.yaml.
5. Modify the capacity key from 1Gi to 2Gi in pvc.ymal file
6. oc create -f pvc.yaml. using oc create tried to create a pvc object, but this also failed with similar error.
7. *Third option:* oc edit pvc jenkins. Tried to modify the capacity from pvc object, but even this failed with error. The error says once the pvc is created it is immutable. SO we cannot resize the pvc.
8. So as all the options are tried out, the only option I see right now is to recreate a new deployment configuration with defined size.
9. Also check with the team if there is any option to resize the existing jenkins pvc.
10. Create a new mount to store the .m2 configuration. As of now by default the .m2 is getting stored at /var/lib/jenkins.
11. The steps would be to create a new mount named maven and mount it to /maven/ i. oc get pvc ii. Create a new pvc object for maven

iii. Using this claim as volume in the pods yaml apiVersion: "v1" kind: "Pod" metadata: name: "mypod" labels: name: "frontendhttp" spec: containers: - name: "myfrontend" image: "nginx" ports: - containerPort: 80 name: "http-server" volumeMounts: - mountPath: "/var/www/html" name: "maven" volumes: - name: "maven" persistentVolumeClaim: claimName: "maven" 3. Modify the maven configuration to store the repositories in the new mount point. 4. Usually maven installed through Jenkins will be placed at <JENKINS\_HOME>/tools.

*references:* <https://docs.openshift.com/enterprise/3.1/dev_guide/persistent_volumes.html>

oc rsync:

1. To copy a single file from the container to the local machine, the form of the command you need to run is:

oc rsync <pod-name>:/remote/dir/filename ./local/dir

1. Uploading files from a conatiner:

oc rsync ./local/dir [pod-name:/remote/dir](http://pod-name/remote/dir)

we can use --exclude=\* and --include=filename and also --delete to have the both folders in sync and also --no-perms to forget about permissions *ref:* <https://blog.openshift.com/transferring-files-in-and-out-of-containers-in-openshift-part-1-manually-copying-files/>

Jenkins Credentails Automation:

There are two options to automate the credentials addition in Jenkins.

*First option:* To store password in plain text in credentials.xml, copy it over to jenkins machine after installing and starting the service. Then jenkins will encrypt it with it's new secret and it will work *Second option:* A second option is to install jenkins, start it and then copy the credentials.xml with encrypted passwords together with secrets directory and secret.xml from previous instllation. This will copy both encryption master key and the encrypted credentials that have been created using this master key.

Copy the credentials.xml file, secrets directory and secrets.xml from working pod to local using rsync.

How to copy the above files to pod as part of automatic deployment.

1. The files should be mounted once the /var/lib/jenkins mount point is available.
2. Store the files in a private git repository in github internal
3. Using pod life cycle hooks, clone the repo on to the pod and also execute a command to copy that to /var/lib/jenkins.*Option 1:* Directly update the dc using oc edit dc jenkins and update the spec with post

Deploy the changes using oc rollout latest jenkins

*Option 2:* Using oc patch from client. create run.sh script and store it in a repo. run.sh contains the below code

source run.sh to run the script to perform a patch operation oc rollout latest jenkins to deploy the latest dc

1. or Use rsync to copy the files as it is time process
2. Can we make use of config maps? Investigate further..

*ref:* <https://blog.openshift.com/using-post-hook-to-initialize-a-database/>

*Failures:*

1. Jenkins persistent is having issues and is going down frequently . So opted for ephermal jenkins image with a volume mount of 100Gi
2. Created a config map for saxon lic and added that as part of the deployment config yaml file,upon rollout deployment was getting failed. Tried troublesshooting but couldn't trace out the issue. The same worked with Persistent image
3. Java/maven is not configured and also no DSL plugin is installed.
4. The idea to install plugins is to create a DSL plugin binary DSL.hpi file from source code and then use jenkins cli upload it to plugins.
5. When tried to install maven using configrue jenkins, maven is not getting installed. So had to manaullay install the maven using oc rsync to copy the binary from local machine to pod and extract it, set maven home and path settings
6. Even java is not configured properly. Need to setup this

Notes:

Most of the times, not able to launch the jenkins url and after sometime it automatically gets up . We don't see any pod/service/route status as down . Find out how to fix this issue

and also need to create a new mount point to store .m2 folder.

dont have access to get the volumes oc get pv

Which volume should be used to create a new mount. How to get the list of volumes and which one we should opt for.