

Publish docker images to Nexus using Jenkins

Install docker in jenkins :

- apt-get update
- apt-get install docker.io -y

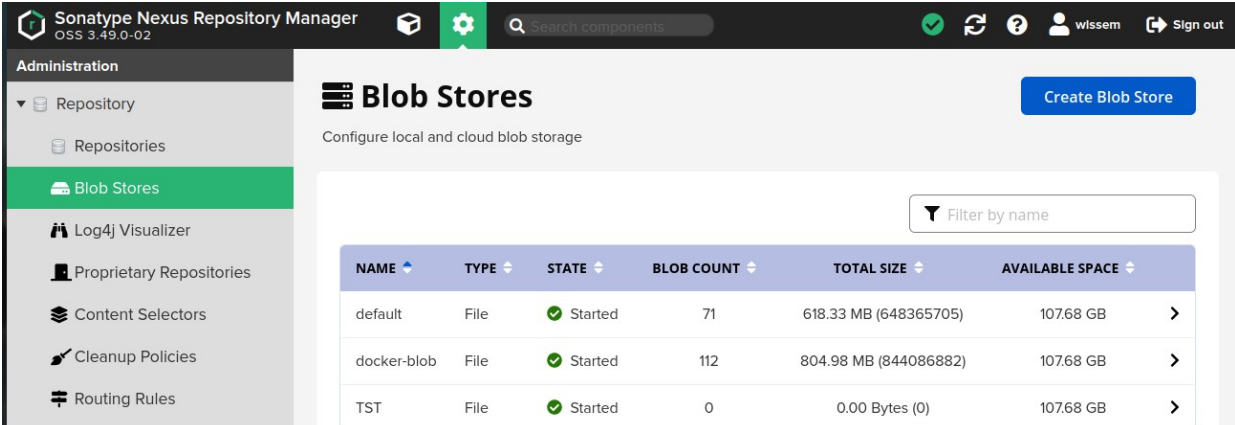
Access Nexus application:

- Get the password from /nexus-data/admin.password files that is present in the nexus container.

```
docker exec -it nexus bash cat /nexus-data/admin.password
```

Blob Store: The blob store is used to store repository contents

- Navigate to *Administration -> Repository -> Blob Stores*.
- Click on *Create Blob Store*: name is docker-blob



The screenshot shows the Sonatype Nexus Repository Manager interface. The left sidebar contains the 'Administration' menu with options like Repository, Repositories, Blob Stores (highlighted), Log4j Visualizer, Proprietary Repositories, Content Selectors, Cleanup Policies, and Routing Rules. The main panel is titled 'Blob Stores' and includes a 'Create Blob Store' button. Below the title, there is a table listing existing blob stores.

NAME	TYPE	STATE	BLOB COUNT	TOTAL SIZE	AVAILABLE SPACE
default	File	Started	71	618.33 MB (648365705)	107.68 GB
docker-blob	File	Started	112	804.98 MB (844086882)	107.68 GB
TST	File	Started	0	0.00 Bytes (0)	107.68 GB

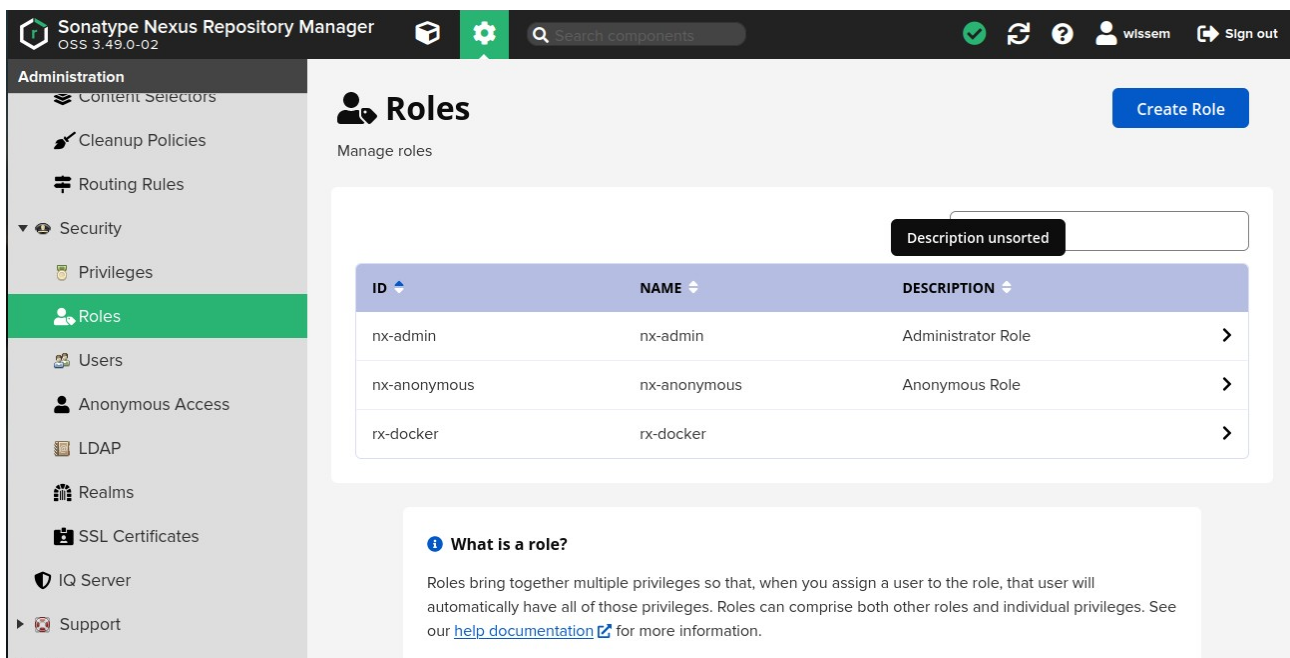
Create a docker repository:create a docker repository on Nexus to publish docker images

- Navigate to *Repositories -> Create repository*.
- Select *docker (hosted)*.
- Name it *dockerhosted-repo*:create an *HTTP* connector at port 8082,Port 8082 is where the docker repository by nexus will be hosted.

- Choose *docker-blob* to store repository contents.

Create a Role: create a role, give access privileges for the *dockerhosted-repo* repository, and then assign this role to a user

- Click on *Create Role*
- Add *Role ID* and *Role Name*. Name as *nx-docker*.
- Add *nexus-repository-view-docker-dockerhosted-repo-** privilege to this role.



The screenshot shows the Sonatype Nexus Repository Manager interface. The left sidebar contains the 'Administration' menu with options like Content Selectors, Cleanup Policies, Routing Rules, Security, Privileges, Roles (highlighted), Users, Anonymous Access, LDAP, Realms, SSL Certificates, IQ Server, and Support. The main area is titled 'Roles' with a subtitle 'Manage roles' and a 'Create Role' button. A table lists existing roles:

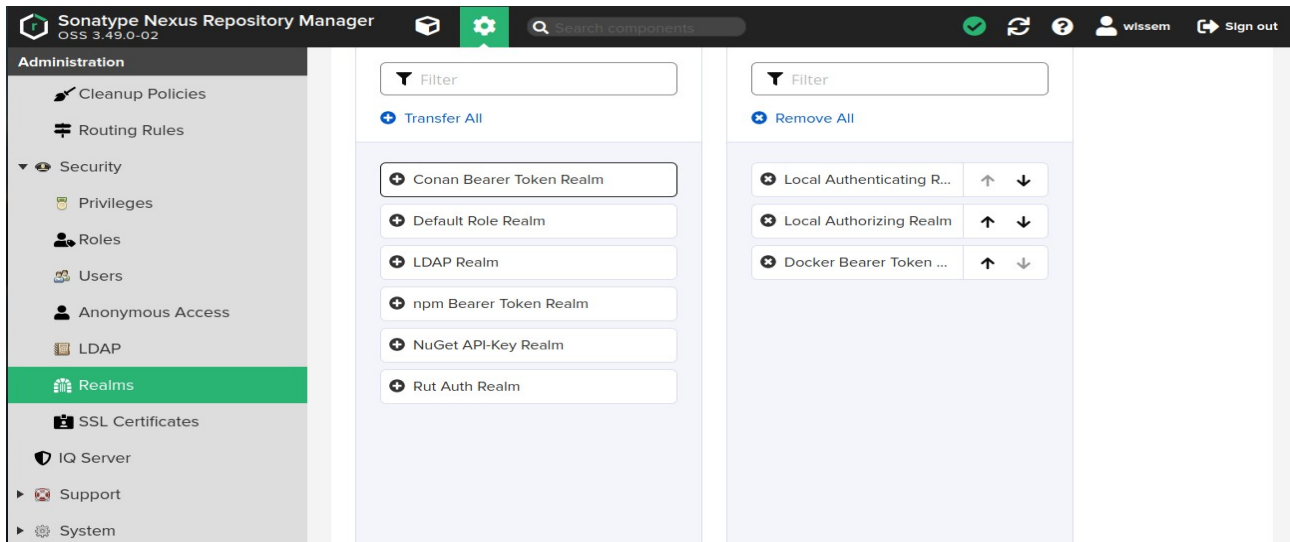
ID	NAME	DESCRIPTION
nx-admin	nx-admin	Administrator Role
nx-anonymous	nx-anonymous	Anonymous Role
nx-docker	nx-docker	

Below the table, there is a section titled 'What is a role?' with explanatory text: 'Roles bring together multiple privileges so that, when you assign a user to the role, that user will automatically have all of those privileges. Roles can comprise both other roles and individual privileges. See our [help documentation](#) for more information.'

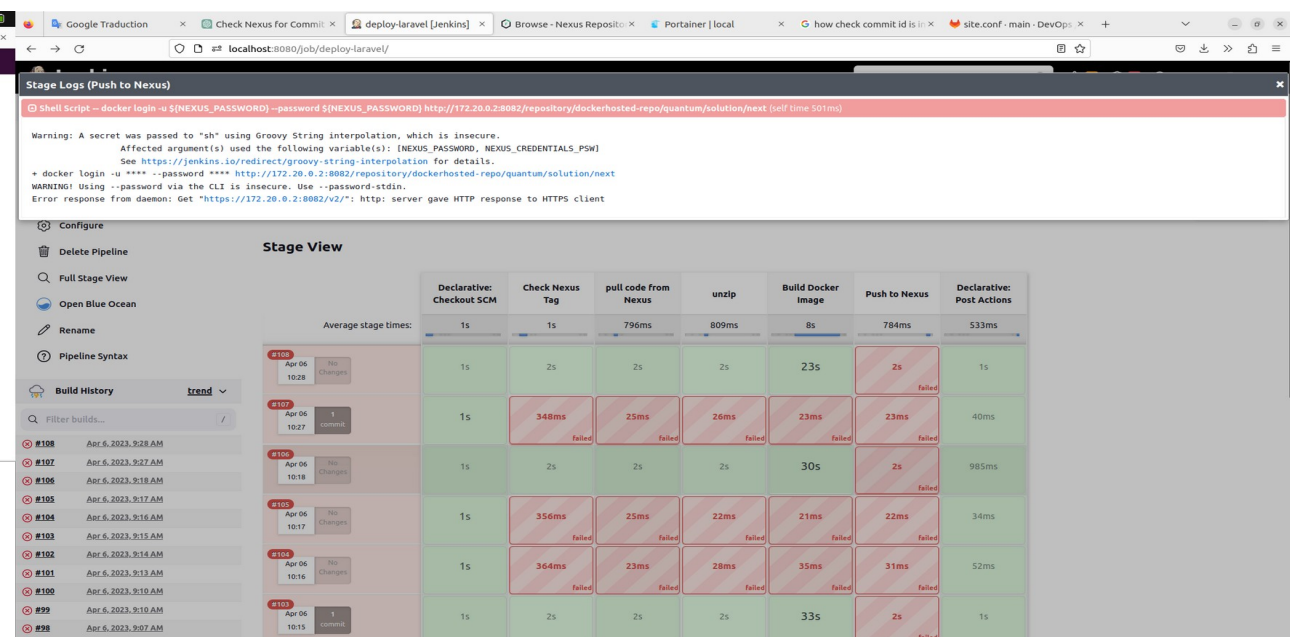
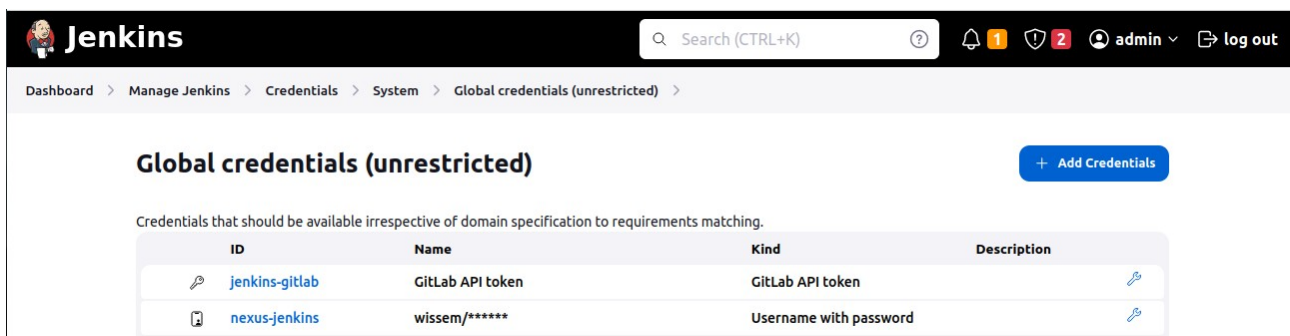
Create a User :

- *Create local user* : name is *wissem* and password is *wissem*

Realms: enable the *Docker Bearer Token Realm* as generally outlined in Realms



Add credentials:



The *dockerhosted-repo* is insecure, it doesn't use *HTTPS* protocol and docker daemon is not letting us access this insecure repository because it might be harmful for our system.

To add the *dockerhosted-repo* to the list of insecure registries, follow the below steps.

- `vi /etc/docker/daemon.json`
- `{ "insecure-registries": ["172.20.0.10:8082"] }`
- `systemctl daemon-reload`
- `systemctl restart docker`
- `docker info`

```
Cgroup Driver: systemd
Cgroup Version: 2
Plugins:
  Volume: local
  Network: bridge host ipvlan macvlan null overlay
  Log: awslogs fluentd gcplogs gelf journald json-file local logentries splunk syslog
Swarm: inactive
Runtimes: io.containerd.runc.v2 runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 2806fc1057397dbaefbea0e4e17bddfdb388f38
runc version: v1.1.5-0-gf19387a
init version: de40ad0
Security Options:
  apparmor
  seccomp
   Profile: builtin
cgroupns
Kernel Version: 5.19.0-38-generic
Operating System: Ubuntu 22.04.2 LTS
OSType: linux
Architecture: x86_64
CPUs: 8
Total Memory: 23.18GiB
Name: wissem-pc
ID: b17b3f4f-e6a2-4052-a36f-de43b176f721
Docker Root Dir: /var/lib/docker
Debug Mode: false
Registry: https://index.docker.io/v1/
Experimental: false
Insecure Registries:
  172.20.0.10:8082
  127.0.0.0/8
Live Restore Enabled: false

root@6b985fece614:/etc/docker#
```

Build the pipeline :



Docker hosted Repository on Nexus

