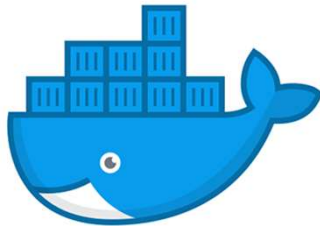
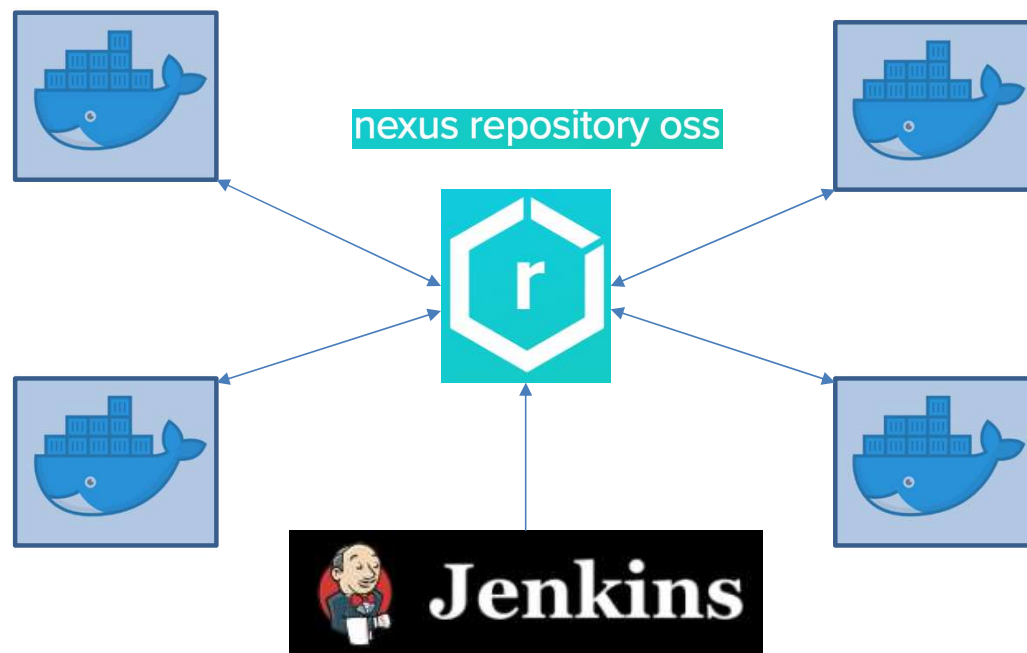


Docker Registry Server

Nexus Sonatype



Docker Registry Server



Nexus Sonatype

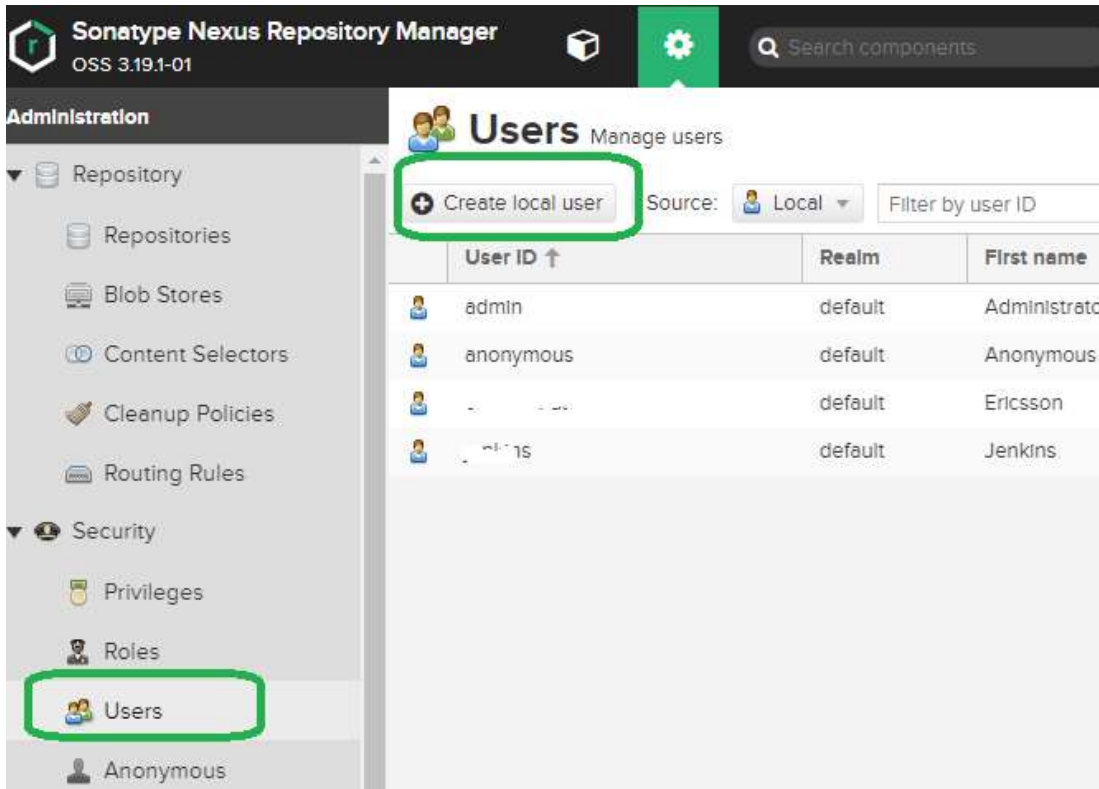
- Free Artifact Repository with support for various types of binary artifacts supporting various build tools → Maven/Java, Docker, Nuget, NPM, APT, YUM and many more
- <https://www.sonatype.com/nexus-repository-oss>
- Needs JRE or JDK to run

>nexus.exe /run

Sonatype Nexus

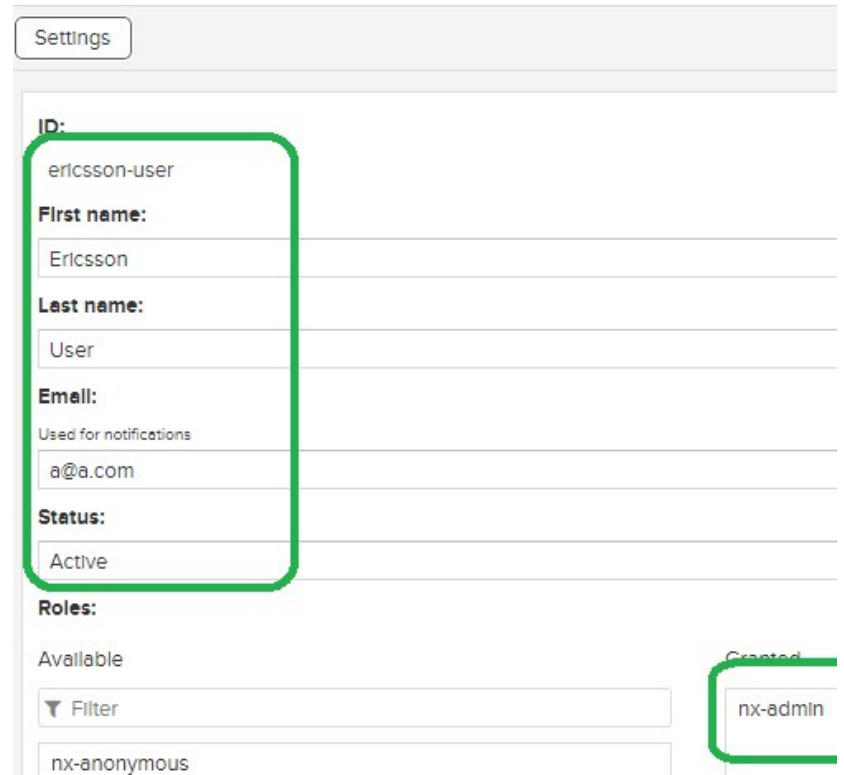
- By default runs on port 8081, change it to run on 8090
 - Modify “Nexus\nexus-3.19.1-01\etc\nexus-default.properties” to change default port if needed
 - Binary to run Nexus.exe : Nexus\nexus-3.19.1-01\bin folder
- >nexus.exe /run
- <http://localhost:8090>
 - Default Credentials : admin/admin123

Create User



The screenshot shows the Sonatype Nexus Repository Manager Administration interface. The left sidebar contains the 'Administration' menu with 'Users' highlighted. The main content area is titled 'Users' and includes a 'Create local user' button, a 'Source' dropdown set to 'Local', and a 'Filter by user ID' input. Below these is a table of existing users.

User ID ↑	Realm	First name
admin	default	Administrato
anonymous	default	Anonymous
ericsson-user	default	Ericsson
jenkins	default	Jenkins



The screenshot shows the user creation form in Sonatype Nexus Repository Manager. The form fields are: ID (ericsson-user), First name (Ericsson), Last name (User), Email (a@a.com), Status (Active), and Roles (Available). A 'Filter' dropdown is also present. The 'Created' field shows 'nx-admin'.

ID: ericsson-user

First name: Ericsson

Last name: User

Email: a@a.com

Status: Active

Roles: Available

Filter: nx-admin

Created: nx-admin

Create User

- User created with following credentials:

User Name : ericsson-user

Password : admin123

Create Docker Hosted Repo

Name:	ericsson-registry
Format:	docker
Type:	hosted
URL:	http://localhost:8090/repository/ericsson-registry/
Online:	<input checked="" type="checkbox"/> If checked, the repository accepts Incoming requests

Ψ CRUDS INFOTECH NG

Repository Connectors

Connectors allow Docker clients to connect directly to hosted registries, but are not always required. Consult our [documentation](#) for which connector is appropriate for your use case. For information on scaling the repositories see our [scaling documentation](#).

HTTP:
Create an HTTP connector at specified port. Normally used if the server is behind a secure proxy.
<input checked="" type="checkbox"/> 8091
HTTPS:
Create an HTTPS connector at specified port. Normally used if the server is configured for https.
<input type="checkbox"/>

Create Docker Hosted Repo

Allow anonymous docker pull:

☒ Allow anonymous docker pull (Docker Bearer Token Realm required)

Docker Registry API Support

Enable Docker V1 API:

☒ Allow clients to use the V1 API to interact with this repository

Storage

Blob store:

default

Strict Content Type Validation:

☒ Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format

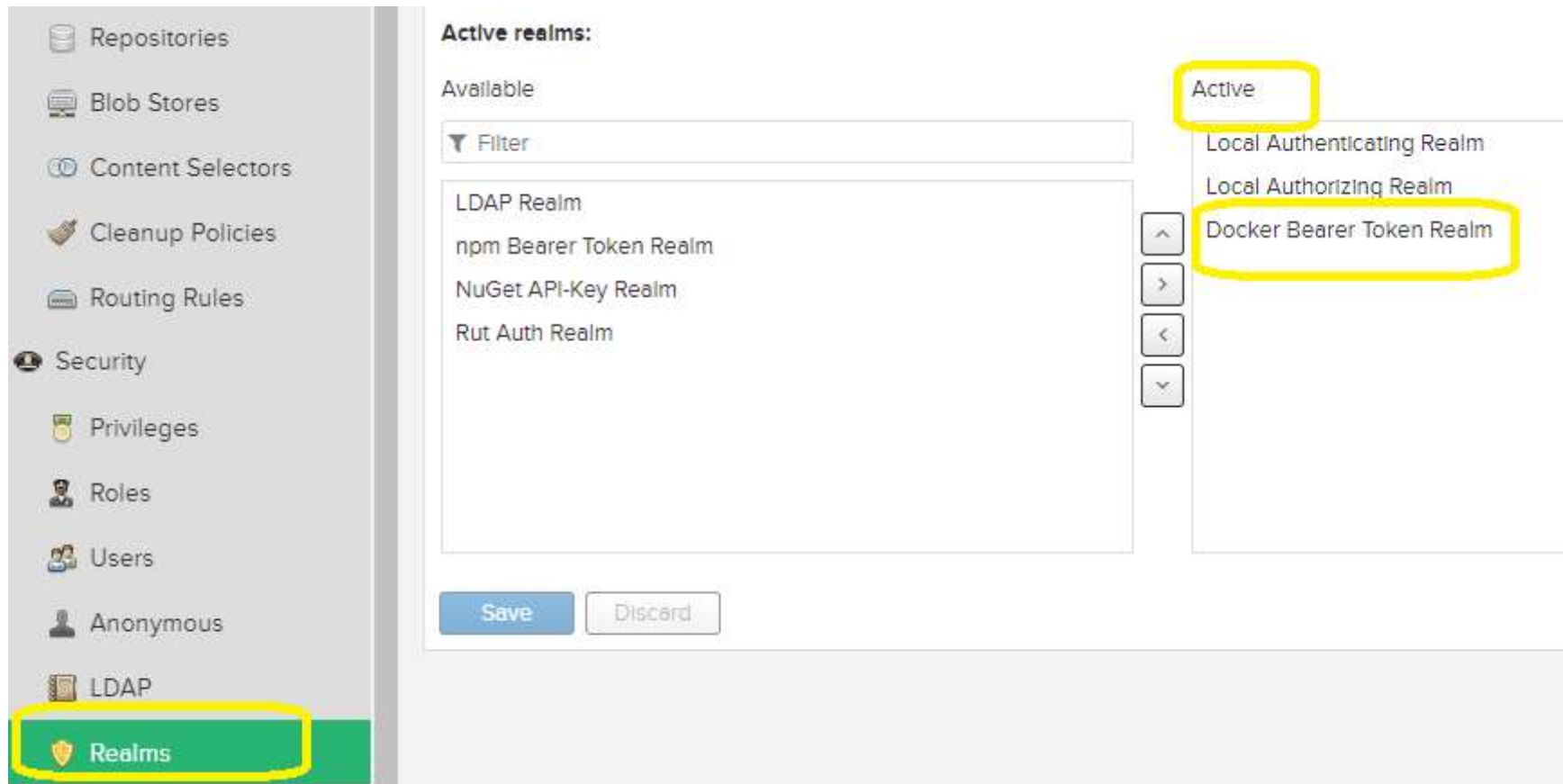
Hosted

Deployment policy:

Controls if deployments of and updates to artifacts are allowed

Allow redeploy

Activate Docker Token Realms



Docker daemon.json

- Prepare docker daemon to access this insecure Registry by adding the following content in /etc/docker/daemon.json

>sudo vi /etc/docker/daemon.json

```
{  
  "insecure-registries" : [ "IP_ADDRESS_OF_NEXUS_SERVER:PORT" ]  
}
```

>sudo systemctl daemon-reload

>sudo systemctl restart docker

Docker Registry

- Check if we can login to the docker registry

```
>docker login -u user-id -p password NEXUS_SERVER:PORT
```

```
>docker login -u ericsson-user -p admin123  
73fb339c8064.ngrok.io
```

```
ubuntu@ip-172-16-1-52:~$ docker login -u ericsson-user -p admin123 73fb339c8064.ngrok.io  
WARNING! Using --password via the CLI is insecure. Use --password-stdin.  
WARNING! Your password will be stored unencrypted in /home/ubuntu/.docker/config.json.  
Configure a credential helper to remove this warning. See  
https://docs.docker.com/engine/reference/commandline/login/#credentials-store  
  
Login Succeeded
```

Docker Image Tag & Push

- How to tag images in-order to push it to our Nexus Registry Server??