Jenkins Project

Installed Jenkins (and Java) on Ubuntu EC2 Instance using the following documentation:
https://www.jenkins.io/doc/book/installing/linux/#debianubuntu

```
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 53 not upgraded.
Need to get 89.2 MB of archives.
After this operation, 90.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd6
4 1.60+git20181103.0eebece-1ubuntu5 [204 kB]
Get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.426.1 [89.0 MB]
Fetched 89.2 MB in 8s (11.2 MB/s)
Selecting previously unselected package net-tools.
(Reading database ... 80296 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../jenkins_2.426.1_all.deb ...
Unpacking jenkins (2.426.1) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Setting up jenkins (2.426.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /lib/s
ystemd/system/jenkins.service.
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-192-0-0-49:~$
```

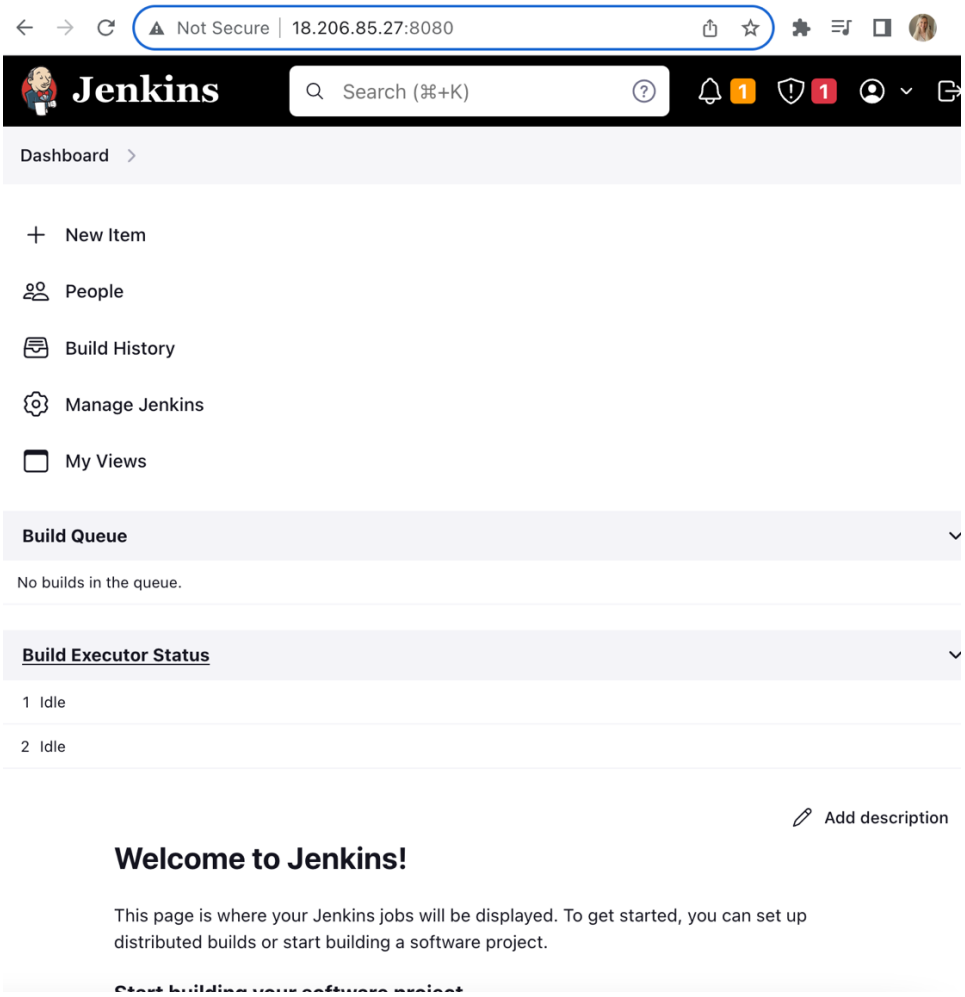
sudo systemctl status Jenkins (you can see the admin password when you run this command)

```
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-11-15 21:22:05 UTC; 1min 54s ago
   Main PID: 5085 (java)
      Tasks: 39 (limit: 1121)
     Memory: 299.3M
        CPU: 38.335s
     CGroup: /system.slice/jenkins.service
             └─5085 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=>
```

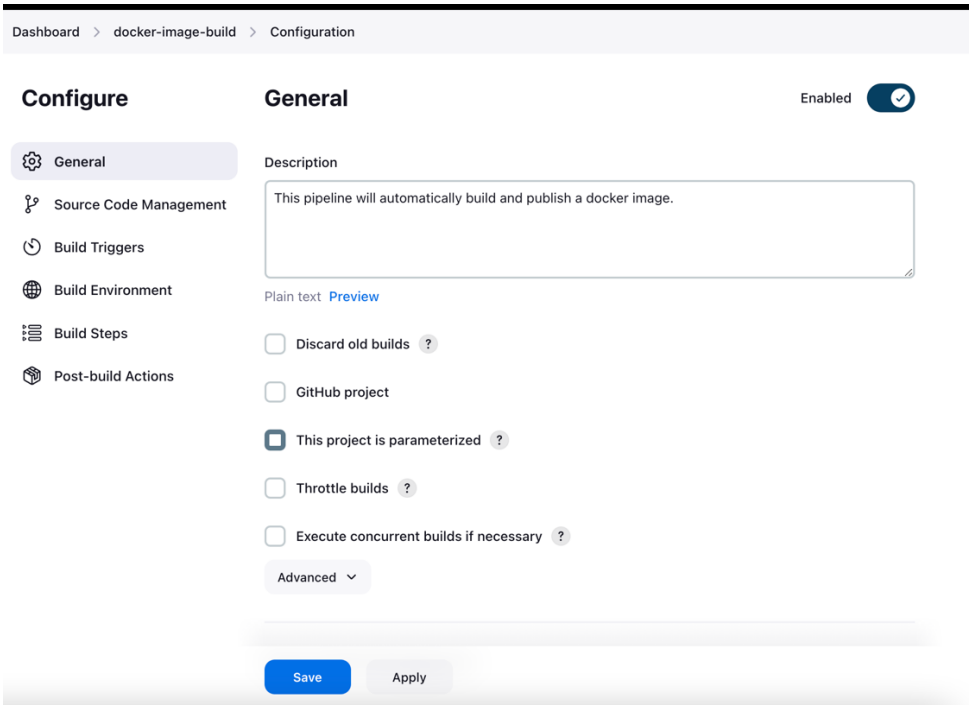
Opening port 8080 and SSH port 22 on the instance firewall:

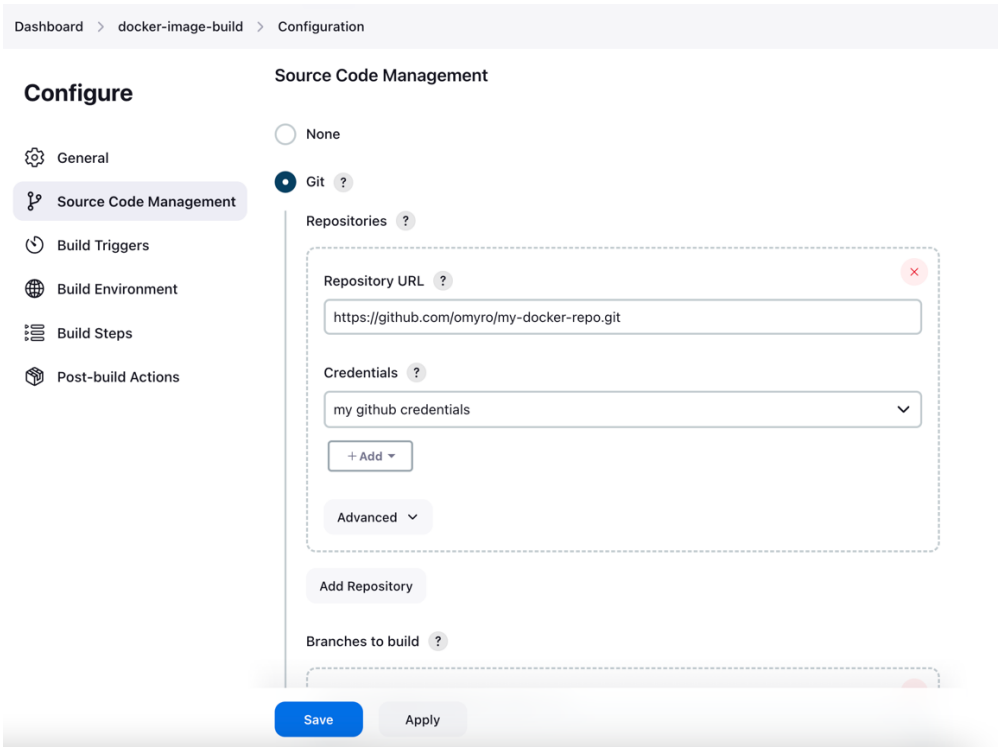
```
ubuntu@ip-192-0-0-49:~$ sudo ufw enable
Firewall is active and enabled on system startup
ubuntu@ip-192-0-0-49:~$ sudo ufw status
Status: active
ubuntu@ip-192-0-0-49:~$ sudo ufw allow 8080
Rule added
Rule added (v6)
ubuntu@ip-192-0-0-49:~$ sudo ufw allow openSSH
Rule added
Rule added (v6)
ubuntu@ip-192-0-0-49:~$
```

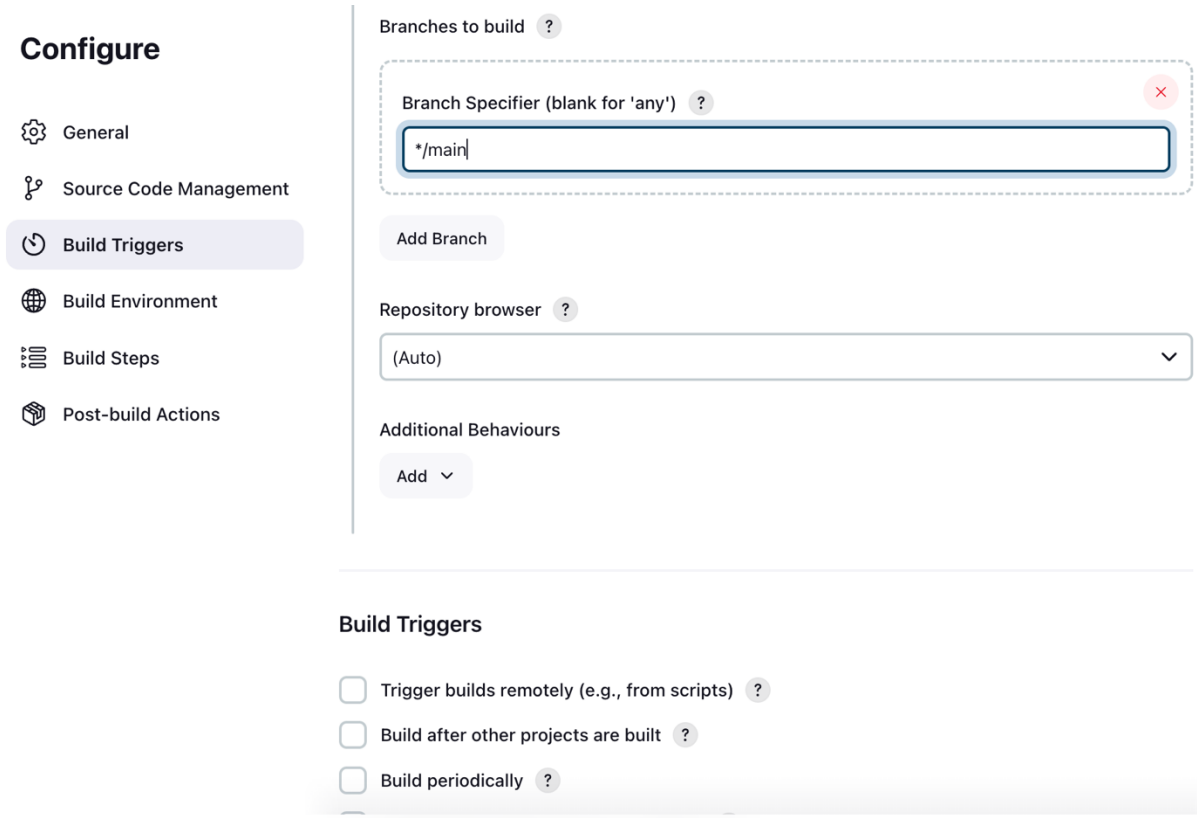
Accessed Jenkins in the browser through port 8080 and following set up instructions:

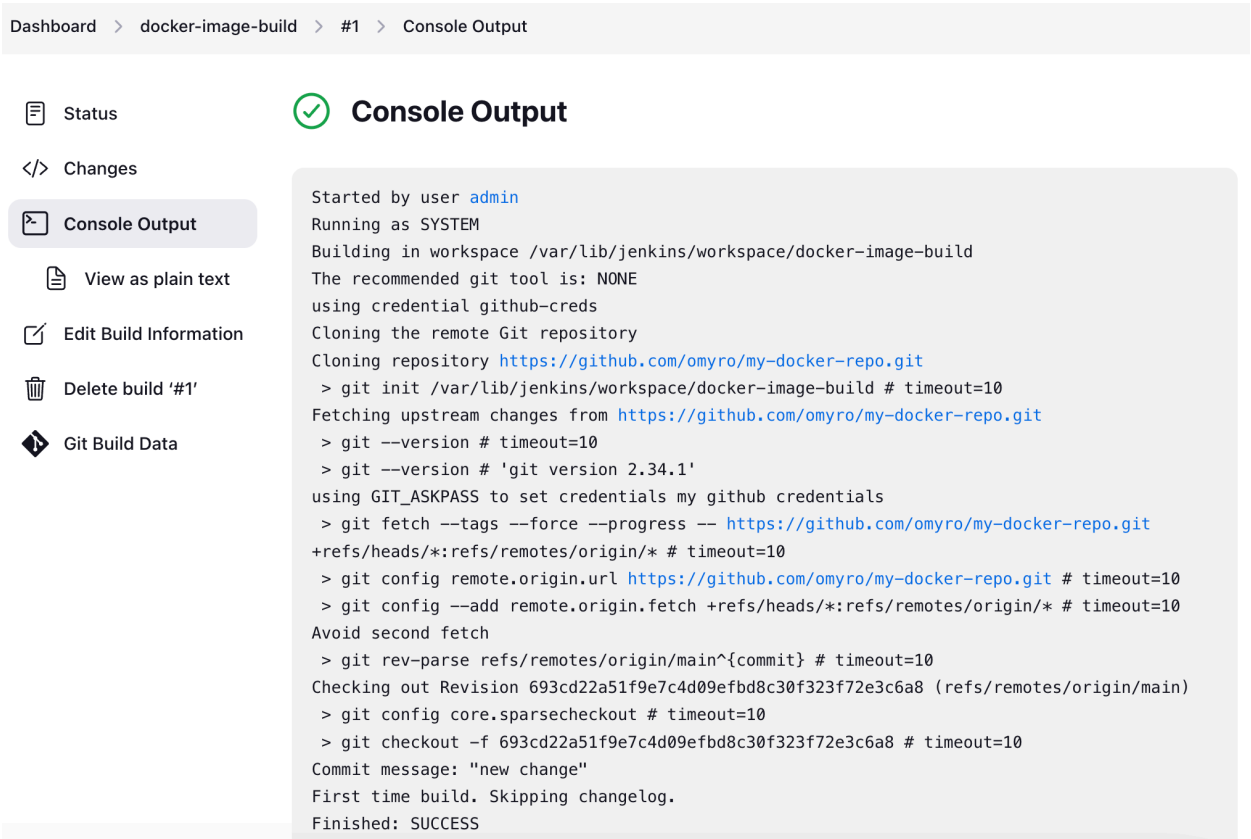


Creating a pipeline to build and publish a docker image to DockerHub:

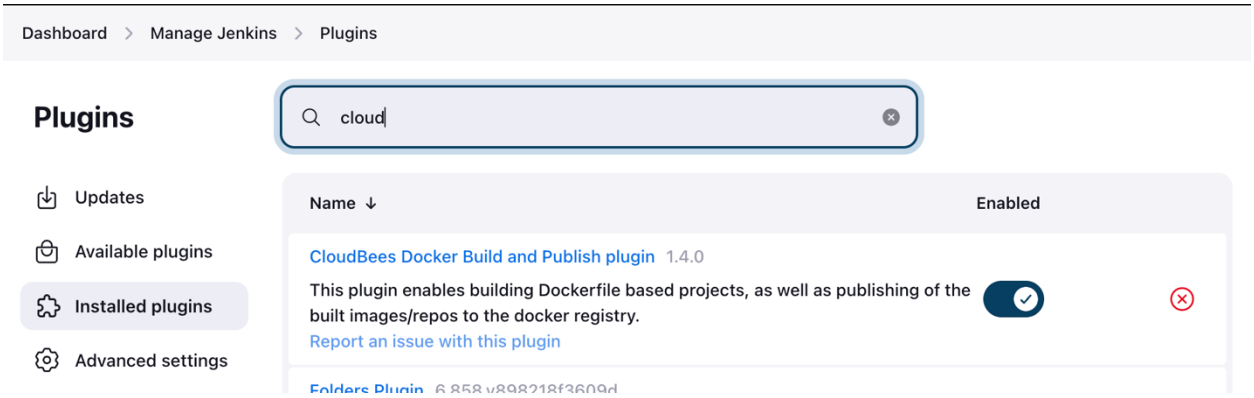

Dashboard > docker-image-build > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
General
Enabled
Description
This pipeline will automatically build and publish a docker image.
Plain text Preview
Discard old builds
GitHub project
This project is parameterized
Throttle builds
Execute concurrent builds if necessary
Advanced
Save
Apply


Dashboard > docker-image-build > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Source Code Management
None
Git
Repositories
Repository URL
https://github.com/omyro/my-docker-repo.git
Credentials
my github credentials
+ Add
Advanced
Add Repository
Branches to build
Save
Apply


Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Branches to build
Branch Specifier (blank for 'any')
*/main
Add Branch
Repository browser
(Auto)
Additional Behaviours
Add
Build Triggers
Trigger builds remotely (e.g., from scripts)
Build after other projects are built
Build periodically


Dashboard > docker-image-build > #1 > Console Output
Status
Changes
Console Output
View as plain text
Edit Build Information
Delete build '#1'
Git Build Data
Console Output
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/docker-image-build
The recommended git tool is: NONE
using credential github-creds
Cloning the remote Git repository
Cloning repository https://github.com/omyro/my-docker-repo.git
> git init /var/lib/jenkins/workspace/docker-image-build # timeout=10
Fetching upstream changes from https://github.com/omyro/my-docker-repo.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
using GIT_ASKPASS to set credentials my github credentials
> git fetch --tags --force --progress -- https://github.com/omyro/my-docker-repo.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/omyro/my-docker-repo.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 693cd22a51f9e7c4d09efbd8c30f323f72e3c6a8 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 693cd22a51f9e7c4d09efbd8c30f323f72e3c6a8 # timeout=10
Commit message: "new change"
First time build. Skipping changelog.
Finished: SUCCESS

Installed the Docker Build and Publish plugin on Jenkins:



Installed docker on the Jenkins server using the following documentation:
https://docs.docker.com/engine/install/ubuntu/

```
ubuntu@ip-192-0-0-49:~$ docker --version
Docker version 24.0.7, build afdd53b
```

Used the sudo usermod -a -G docker ubuntu command and the sudo usermod -a -G docker jenkins command to give permissions to these users. Exit to logout and then log back in. Now docker info works without the sudo in front.

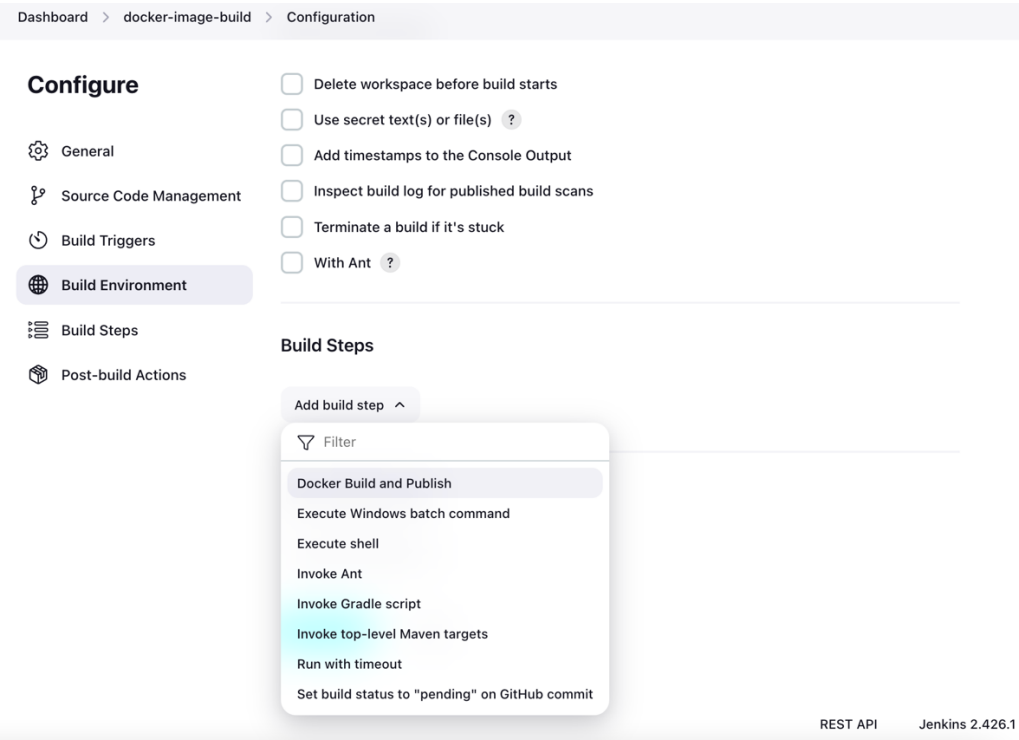
```
$ sudo su - ubuntu
ubuntu@ip-192-0-0-49:~$ docker info
Client: Docker Engine - Community
 Version:    24.0.7
 Context:    default
 Debug Mode: false
 Plugins:
  buildx: Docker Buildx (Docker Inc.)
    Version:  v0.11.2
    Path:     /usr/libexec/docker/cli-plugins/docker-buildx
  compose: Docker Compose (Docker Inc.)
    Version:  v2.21.0
    Path:     /usr/libexec/docker/cli-plugins/docker-compose

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 24.0.7
 Storage Driver: overlay2
```

Need to restart Jenkins for the jenkins user permission to take effect:

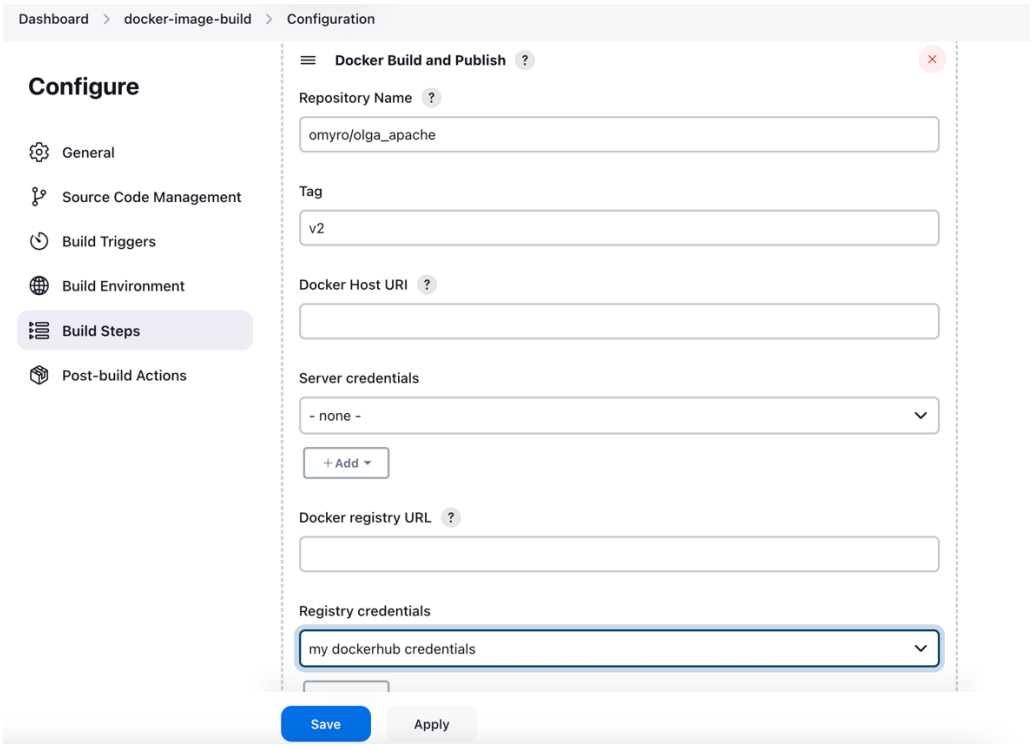
```
ubuntu@ip-192-0-0-49:~$ sudo systemctl restart jenkins
```

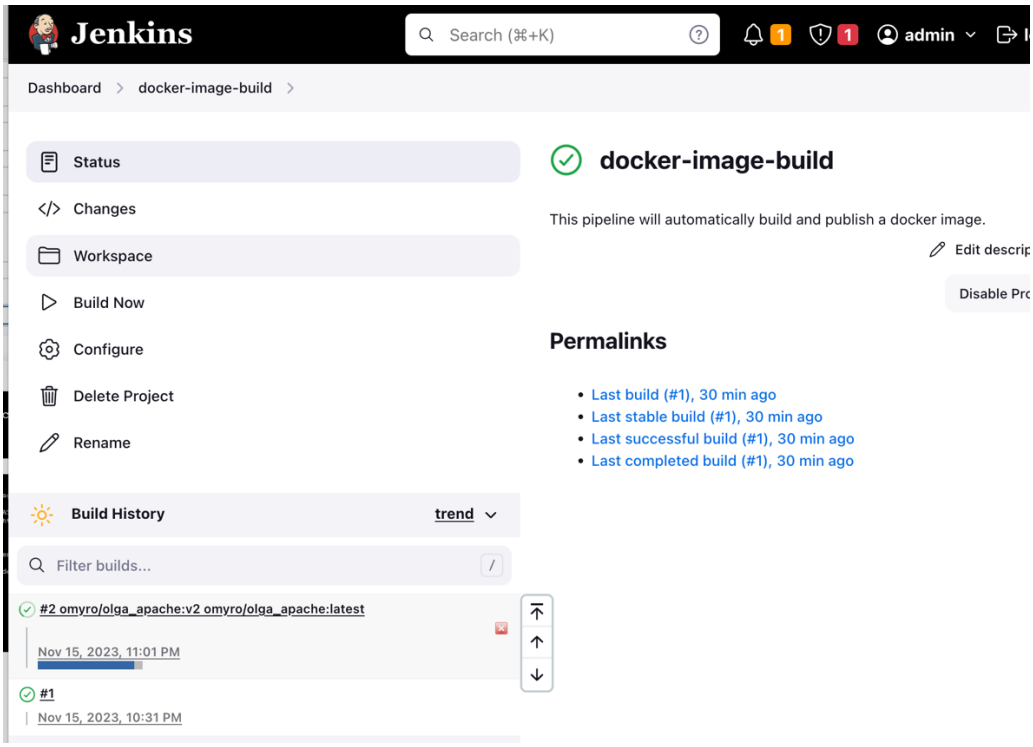
Need to log into docker on server for the build to work properly:

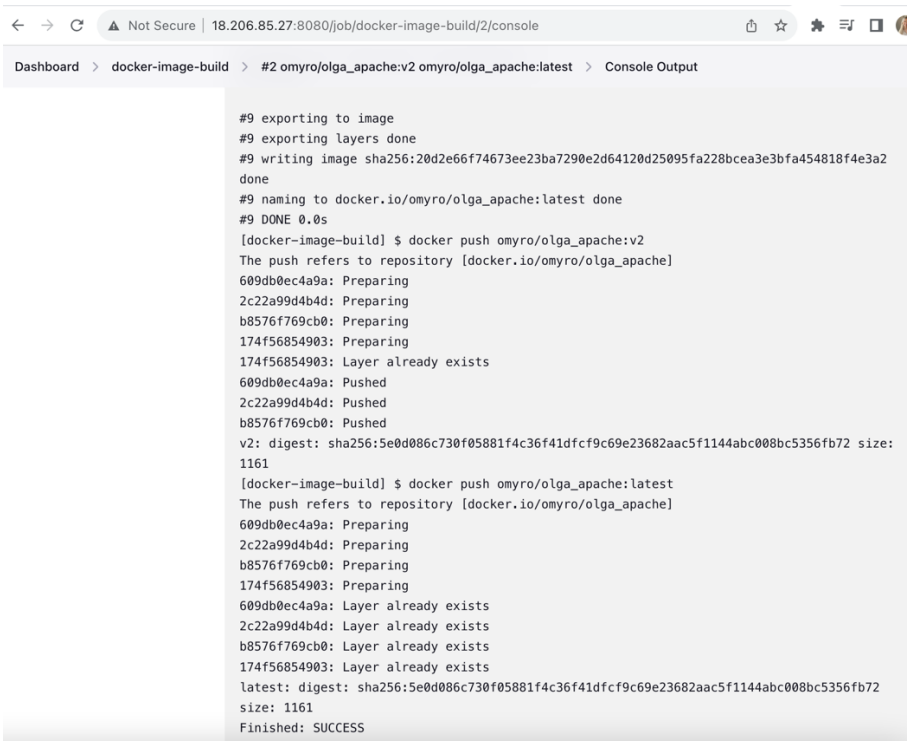
```
ubuntu@ip-192-0-0-49:~$ sudo su - jenkins
jenkins@ip-192-0-0-49:~$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker I
D, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better secur
ity and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username: omyro
Password:
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
jenkins@ip-192-0-0-49:~$
```



Creating v2 of the olga_apache docker image using the Jenkins pipeline:







Shows SUCCESS – Published version 2 of the olga_apache docker image to DockerHub using the Jenkins pipeline!!!