

Integrating GitHub with Jenkins on AWS EC2 (Ubuntu)

Saturday, March 16, 2024

11:16 PM

1. Create an Ubuntu VM on any cloud (AWS) with port number
 - a. 80
 - b. 443
 - c. 8080 (for Jenkins application)
 - d. 22
2. Connect to the EC2 instance using SSH via
 - a. Putty
 - b. MT Putty
 - c. MobaXterm
 - d. PowerShell
 - e. Windows Terminal
3. Update the ubuntu server and install docker on it.
 - a. Set up Docker's apt repository

```
# Add Docker's official GPG key:
```

```
sudo apt-get update -y
```

```
sudo apt-get install ca-certificates curl -y
```

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
```

```
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
# Add the repository to Apt sources:
```

```
echo \
```

```
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
```

```
https://download.docker.com/linux/ubuntu \
```

```
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
```

```
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update -y
```

- b. Install docker packages

```
apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin -y
```

- c. Verify that the Docker Engine installation is successfully

```
docker run hello-world
```

```
# systemctl enable docker
```

```
# systemctl start docker
```

```
# usermod -a -G docker <ubuntu>
```

- d. Pulling Jenkins image from dockerhub.

```
# docker pull jenkins/jenkins
```

- e. Listing images:

```
# docker images
```

- f. Create a new directory:

```
# mkdir Jenkins
```

- g. Running Jenkins container on port 8080

```
# docker run -d --name jenkins -p 8080:8080 -v $PWD/jenkins/ jenkins/jenkins
```

```
# docker ps
```

- h. Login to <EC2-INSTANCE-PUBLIC-IP>:<PORT-NUMBER>

```
http://<ec-instance>:<8080>
```

```
Unlock the Jenkins using
```

```
# docker ps
```

```
# docker exec -it <e580724f6f66> cat /var/jenkins_home/secrets/initialAdminPassword
```

```
cat $(cat /dev/urandom | tr -dc 'a-z0-9' | fold -w 64 | xargs sha1sum | cut -d ' ' -f 1) | xargs echo | sed 's/\"/>

```

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

```
/var/jenkins_home/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

Copy this password and paste it to browser page.

Select **install suggested plugins**:

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Wait for it:

Getting Started

Getting Started

☒ Folders

☐ Timestampers

☐ Pipeline

☐ Git

☐ LDAP

☒ OWASP Markup Formatter

☐ Workspace Cleanup

☐ GitHub Branch Source

☐ SSH Build Agents

☐ Email Extension

☒ Build Timeout

☐ Ant

☐ Pipeline: GitHub Groovy Libraries

☐ Matrix Authorization Strategy

☐ Mailer

☐ Credentials Binding

☐ Gradle

☐ Pipeline: Graph View

☐ PAM Authentication

☐ Dark Theme

Installed Plugins

++ Jenkins API

++ Folders

++ OWASP Markup Formatter

++ Ant

++ Pipeline: Step API

++ Pipeline: Groovy

++ Build Timeout

++ Credentials

++ Plain Credentials

++ Git

++ Trilead API

Fill the below details:

Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

Save & continue.

Check the URL: <http://<IP-OF-EC2-INSTANCE>:8080/>

Getting Started

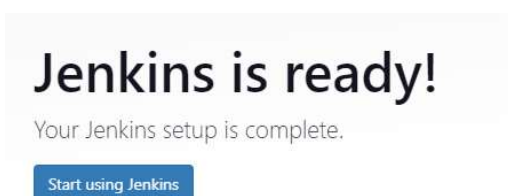
Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.

The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Save & finish.



Switch to GitHub & create a new public repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner * / Repository name *
jenkinsgitrepo is available.

Great repository names are short and memorable. Need inspiration? How about [fantastic-octo-umbrella](#)?

Description (optional)

- ☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template:

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

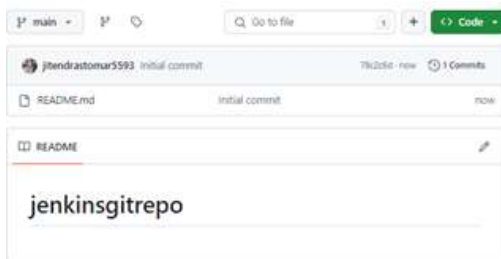
License:

A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

① You are creating a public repository in your personal account.

Create repository



On GitHub, go to “settings” & then “webhooks”. Click “Add webhook”.

Webhooks Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

Payload URL *

Content type

Secret

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me everything.

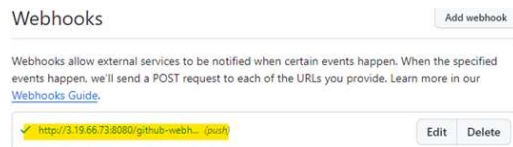
☐ Let me select individual events.

Payload URL → <http://<jenkins-svr-public-IP>:<port>/github-webhook/>

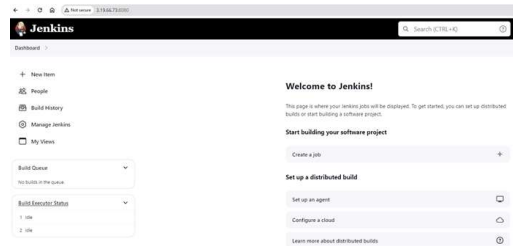
Content type → application/json

Which events would you like to trigger this webhook → Just the push event

Ensure “Active” is check marked. & click on “Add webhook”. & wait for some time.



Now switch back Jenkins portal, Jenkins dashboard: - Create “new item”.

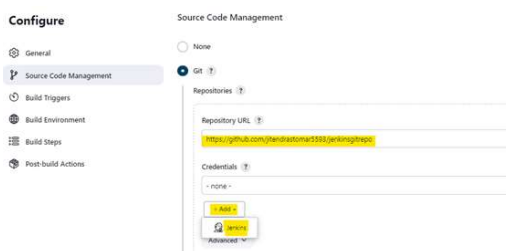


Enter an item name:

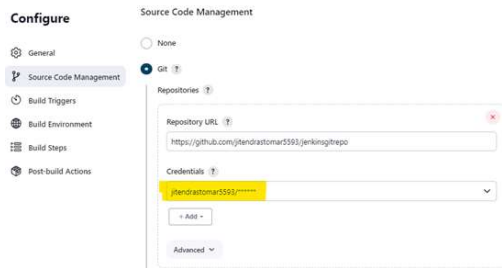


Click OK.

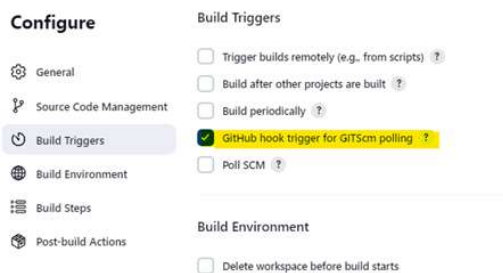
Under “Configure” select “source code management”, select GIT & fill ‘repository URL’.



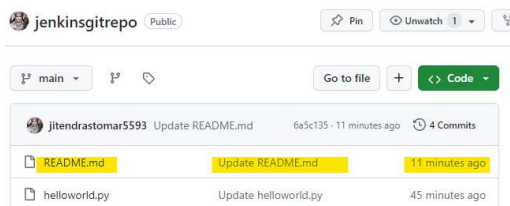
Select “Jenkins” & under credentials, select username & password & fill those.



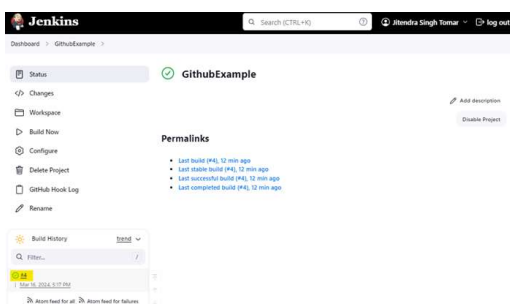
Click on “Build Triggers”. Select GitHub hook.

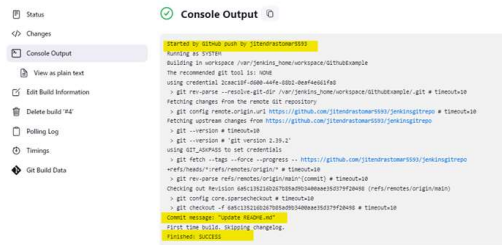


Click “Build Steps”, Nothing to select and save. Switch back to GitHub & edit the README.md file & commit it.



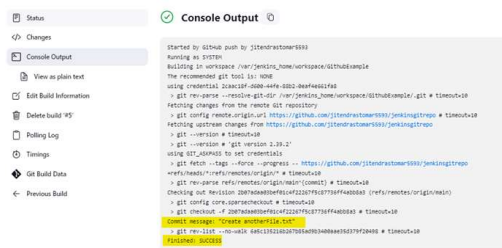
Go back to jenkins dashboard page & wait for build history to show up, automatically.





```
Started by GitHub push by jstendraswer5883
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/15thd8sample
The recommended git tool is: NONE
using credential 20ac3cf-006-a4fe-0b2-deaf4e61f6d
> git re-base --rebase-merge --no-ff --no-commit --no-merge --no-ff --no-commit --no-merge
Retrieving changes from the remote git repository
> git config remote.origin.url https://github.com/jstendraswer5883/jenkinsgtrepo # timeout=10
Retrieving upstream changes from https://github.com/jstendraswer5883/jenkinsgtrepo
> git --version # timeout=10
> git --version # git version 2.38.1
using GIT_SSH to set credentials
> git fetch --tags --force --progress -- https://github.com/jstendraswer5883/jenkinsgtrepo
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git re-base refs/remotes/origin/main # timeout=10
Checking out Revision 56c13202a705d4e03408a8e3d79f2405 (refs/remotes/origin/main)
> git config core.sshCommand # timeout=10
> git checkout -f 56c13202a705d4e03408a8e3d79f2405 # timeout=10
Commit Message: "Create README.txt"
First time build. Skipping changelog.
Finished: SUCCESS
```

Creating another file & commit on GitHub & switch to Jenkins dashboard and visit build history.



```
Started by GitHub push by jstendraswer5883
Running as SYSTEM
Building in workspace /var/jenkins_home/workspace/15thd8sample
The recommended git tool is: NONE
using credential 20ac3cf-006-a4fe-0b2-deaf4e61f6d
> git re-base --rebase-merge --no-ff --no-commit --no-merge --no-ff --no-commit --no-merge
Retrieving changes from the remote git repository
> git config remote.origin.url https://github.com/jstendraswer5883/jenkinsgtrepo # timeout=10
Retrieving upstream changes from https://github.com/jstendraswer5883/jenkinsgtrepo
> git --version # timeout=10
> git --version # git version 2.38.1
using GIT_SSH to set credentials
> git fetch --tags --force --progress -- https://github.com/jstendraswer5883/jenkinsgtrepo
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git re-base refs/remotes/origin/main # timeout=10
Checking out Revision 56c13202a705d4e03408a8e3d79f2405 (refs/remotes/origin/main)
> git config core.sshCommand # timeout=10
> git checkout -f 56c13202a705d4e03408a8e3d79f2405 # timeout=10
Commit Message: "Create another file.txt"
First time build. Skipping changelog.
Finished: SUCCESS
```

From <<https://d.docs.live.net/4b07f4230dd85219/Documents/Jenkins%20documentation%20-%20working%20docx.docx>>