Ollscoil na Gaillimhe
University of Galway

Lab 3: CI/CD: Jenkins

Objective:

Installing and using Jenkins for CI/CD

1. Creating a Virtual Environment

1.1. Create a Linux Virtual Machine

Using VirtualBox, create a virtual machine, running Ubuntu 22.04 LTS (https://ubuntu.com/download/desktop).

Minimum VM configuration (you can put more if your machine allows it):

- 2 core CPUs
- 4 GB of RAM
- 30GB of Disk

2. Installing Docker

2.1. Install Java

Once your VM is installed, install Java (the default Java version available).

```
sudo apt update
sudo apt install default-jre
```

You can check what version of Java is installed as below. The Java version should be 11 or higher.

```
java --version
```

Docker (navigate to Get Docker site to access the Docker download that's suitable for your platform)

2.2. Install Docker

Download and install Docker

- sudo apt install apt-transport-https ca-certificates curl software-properties-common
- curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
- echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

- sudo apt update
- apt-cache policy docker-ce
- sudo apt install docker-ce

Checking Docker installation

- sudo systemctl status docker
- `sudo docker version`

Adding a Docker User

To avoid having to type sudo everytime, you can add a Sudo Docker User. In my case, my username is caller ¨takfarinas¨

- sudo usermod -aG docker takfarinas //NOTE: replace takfarinas with whatever is your username

Testing the working of your Docker

You can test if your docker is working properly as follows:

- docker run hello-world

3. Install Jenkins

3.1. Download and Run Jenkins

1. Download Jenkins Generic Java package (.war)
   a. From here-> https://www.jenkins.io/download/
   b. If you do not have a graphical interface, you can use the command "wget" to download it
2. Open up a terminal in the download directory
3. Run the command line below (note: you need to replace /LOCATION/ with the path to the location of jenkins.war, it is likely to be in (~/Downloads/jenkins.war):

```
java -jar /LOCATION/jenkins.war --httpPort=8080
```

3.2. Configure Jenkins

Open a Browser and navigate to http://localhost:8080 and follow the instructions to complete the installation.

You will particularly be prompted to get the default password. It is likely to be in the file /home/USERNAME/.jenkins/secrets/initialAdminPassword.

In a new terminal, you can open the file using gedit using the following command (replace USERNAME with your own username):

```
gedit /home/USERNAME/.jenkins/secrets/initialAdminPassword
```

You will find a key that looks like this: 3a9c4fa684974f5e92b83c4420169164

Then, select ¨Install suggested plugins¨

Create a User Administrator.
When the installation is complete, you can start putting Jenkins to work!

4. Freestyle Project

The most basic way of using Jenkins is through Freestyle projects (i.e., a succession of command lines).
In this type of project, you need to make sure that any command line that you execute exists and that the way you execute it is correct.

4.1 Crating a Freestyle project

To create a free style project from the main Jenkins dashboard:

- Click on New Item
- Enter a project name (e.g., f1)
- Select Freestyle project
- Click Ok

In the general window:

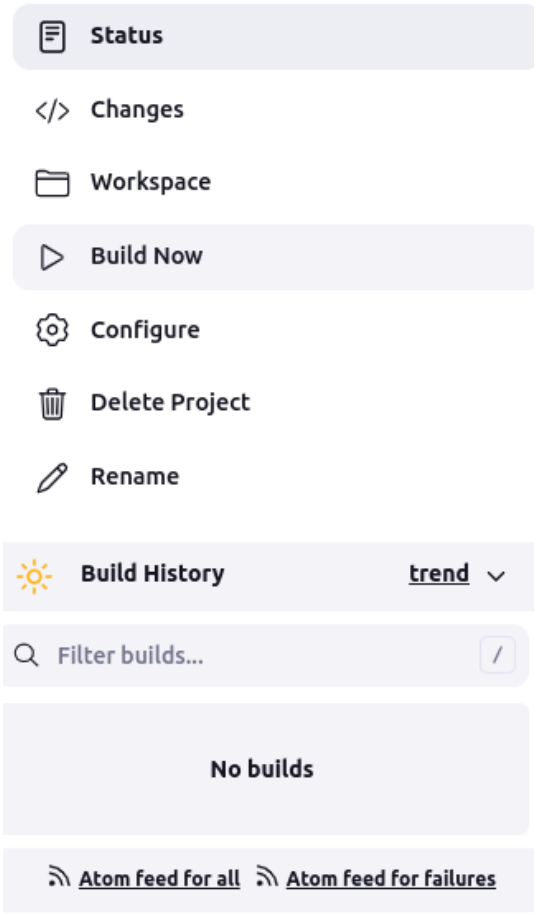
- Write a description (e.g., This is a test of freestyle projects)
- Under ¨Build Steps¨, click on: ¨Add build step¨, you will see several options
    - Select ¨Execute Shell¨
    - Write the script below (or any other bash commands, e.g., Maven goals):

```
ls -al
for i in 1 2 3; do echo "Number $i" >> myFile.txt; done
cat myFile.txt
rm myFile.txt
```

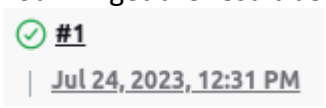
- Save

4.2. Running a Freestyle Project

- Click on Build Now:

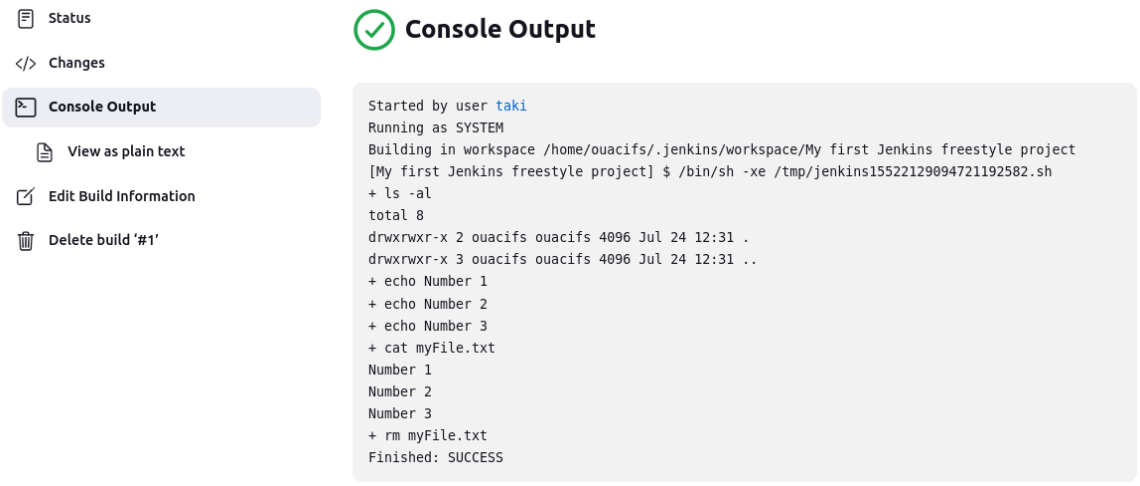


- You will get the result below:



4.3. Viewing the results:

Click on the Results of the build (1)
Then on the “Console Output”, you will see the console output of each step in your previous script.
You will also see at the end a statement that the job has successfully finished



5. Sequencing Freestyle projects

In Jenkins, you can create a sequence of freestyle projects so that you will be prompted to build a project “f2” once the project "f1" is built successfully.

5.1. Create a New Freestyle Project

Create a new Freestyle project “f2”.

This time, in the “Build Triggers” section, make sure to tick on “Build after projects are built”.
In the text box that opens, type “f1, ”.
Then, save.

5.2. Build the First Freestyle Job

Now, build “f1”
You will see that once “f1” is completed, it will prompt you to build “f2”

6. Automation with Freestyle Projects

Jenkins freestyle projects are a first step towards automation. Instead of executing some random Bash commands, you can for instance download a Maven-based Java project from GitHub, clean it, download dependencies, compile it, and run it with Maven.

While Jenkins freestyle projects allows for a wide range of customization in terms of build configurations and automation, they are similar to having a bash script in many ways.

In the next practical, we will use a different type of Jenkins projects: Pipelines

7. Task

- Create a **free-tier** Amazon EC2 VM, t3.micro
  - Running **Ubuntu**
  - 1 GiB Memory
  - 30GiB of Disk
- Make sure to allow HTTP and HTTPS traffic from the internet
- Also, add a custom Inbound connection:
  - Type: TCD
  - Port: 8080
  - From: IPV4 from Anywhere
- Install Jenkins on your VM.
- Use Jenkins (in Amazon EC2) from your computer
  - Use a browser
  - On the Public IP address given by amazon and the Port 8080