Walk-Through: Testing Machine on VirtualBox

Abstract

Jenkins is a continuous integration tool with many extensions that allows to standardize the project lifecycle: it ensures that appropriate revision of the project is used for the current build, that the software is tested accordingly to expected schedule and the testing results are summarized in a user-friendly report. We will use this setup to grade your C++ exercises.

Using Jenkins by students is NOT mandatory. Jenkins is just the place where different tools such as GitHub, shellscripts, Makefiles, crontab scheduling, Web-based reporting can be put together. There are many other reasonable ways how you can integrate these tools – you are encouraged to use whatever works for you.

1 Objectives

Building and testing of student projects should be consistent to align expectations of students and instructors. The following features make Jenkins continuous build software a good match for our C++ class.

Virtual Environment
: Run C++ projects in an environment with standardized operating system and its environment variables. VirtualBox and Xubuntu ensure this standardization.

Directory Layout
: Project files should have predictable layout; version control (such as GitHub) should be used to mark the submission, if possible.

Execution Time
: Jenkins build steps can have timeout limitations so that C++ program execution is stopped, if it goes into an infinite loop or runs an inefficient algorithm.

Scheduling Build Tasks
: Build steps (and also grading of student submissions) can be scheduled using crontab-like notation.

User-Friendly Reporting
: If build tasks fail or tests do not match the expected files, this should be easy to see.

2 Walk-Through Outline

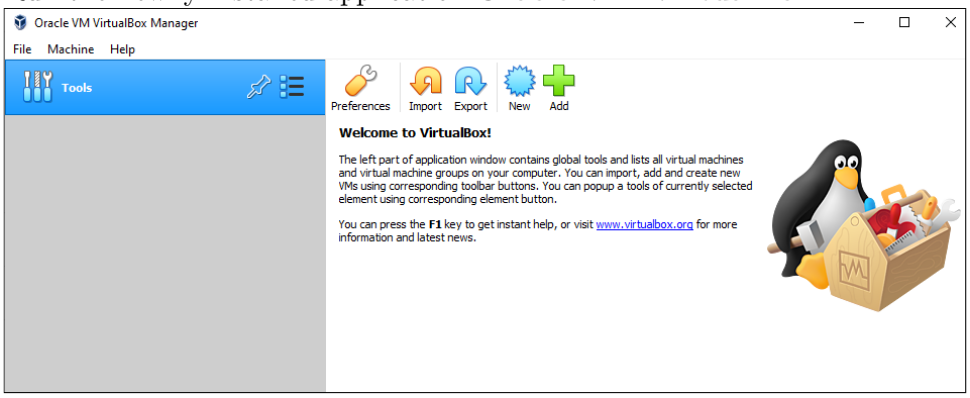
This walk-through will consist of the following major steps (their detailed descriptions follow in the next subsection).

1. Install VirtualBox software. Create a guest machine slot named `Xubuntu` on VirtualBox.
2. Create and configure Xubuntu guest machine.
3. Install basic software on the Xubuntu guest.
4. Set up Jenkins software on Xubuntu.
5. Configure an additional Network Interface Controller (NIC) on guest machine to allow remote connection.
6. Configure a build task on Jenkins and run it.

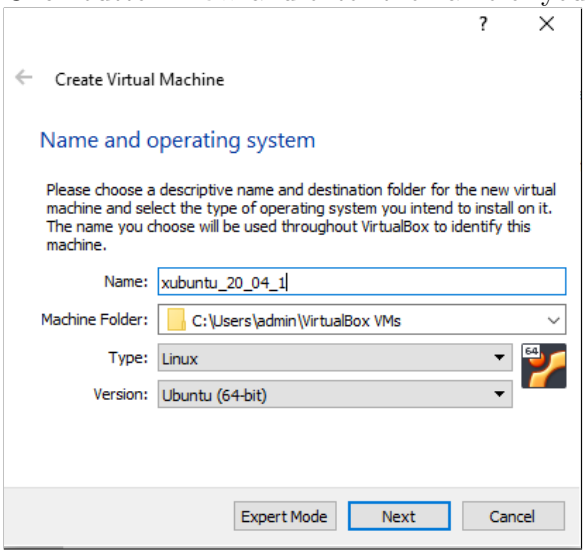
3 Walk-Through Steps

3.1 VirtualBox Setup

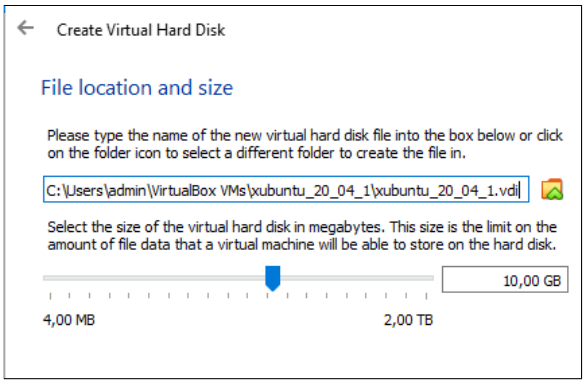
1. Visit `https://www.virtualbox.org/` and download the most recent VirtualBox installer.
   (a) Click on **Download VirtualBox 6.1** banner.
   (b) Click the link **Windows hosts**, if your physical machine is Windows 10 laptop (or choose other operating system – to whatever you have).
   (c) Save the instler, such as **VirtualBox-6.1.12-139181-Win.exe**.
2. Double-click on the VirtualBox installer (elevate privileges to Admin-level, if asked to do so), and pick the default values to install it.
3. Run the newlly installed application **Oracle VM VirtualBox**.



4. Click button **New** and enter the name of your new virtual guest, for example, `xubuntu`.



5. Leave the default RAM memory size (1024 MiB). If your laptop is powerful (16 or more GiB of RAM), consider giving more RAM memory, say, 2048 MiB.
6. Leave the default option **Create a virtual hard disk now**; also leave the **VDI (VirtualBox Disk Image)**.
7. Leave the default option **Dynamically allocated**.
8. Confirm the location of virtual memory image.

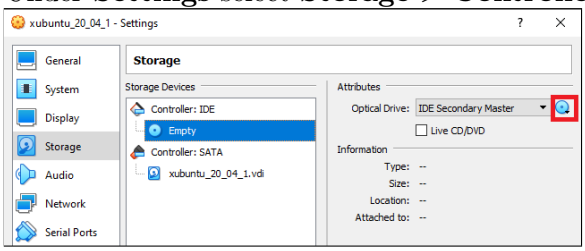


3.2 Creating Xubuntu Guest

1. Download the Xubuntu installer (as an ISO file of some stable release). Visit `https://xubuntu.org/download/` and pick a 64-bit ISO image. In our example it is **xubuntu-20.04.1-desktop-amd64.iso**.

2. Make sure that the guest machine is powered off, select `xubuntu` machine and click button **Settings**.

3. Under **Settings** select **Storage** > **Controller IDE** > **Empty**.



4. Click on the browse button (highlighted in red in the above image). Select the Xubuntu image that you downloaded earlier.

5. In the VirtualBox application, select the `xubuntu` machine and click on the button **Run** (the green arrow).

6. Wait about 5 minutes until Xubuntu image loads from the virtual CD-ROM drive. Click on the button **Install Xubuntu**.

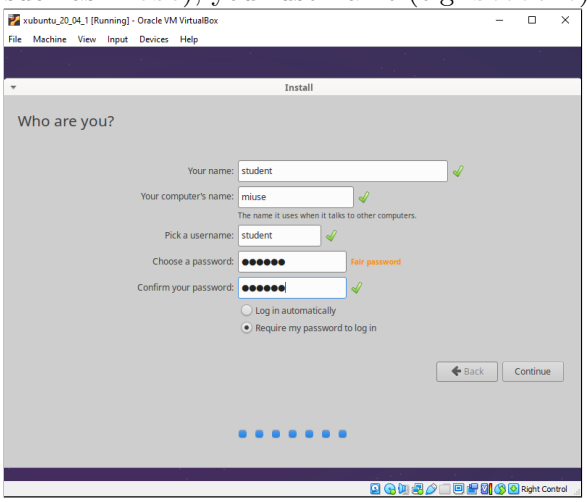
7. Leave the default keyboard layout **English (US)** > **English (US)**.

8. Selecting the checkbox **Select third party software...** in the Xubuntu installer is optional (it is selected on instructor machines).

9. Leave the default radio button **Erase disk and install Xubuntu**.

10. Select **Riga** as your current location.

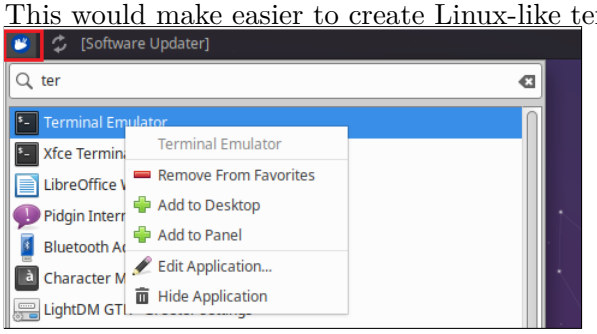
11. Enter an Xubuntu Linux machine name (some short name with lower-case English letters such as `miuse`), your username (e.g. `student`) and some password (e.g. `Bitl1!`).



Note. At this point you would need to wait about 15 minutes until VirtualBox finishes installing Xubuntu guest.

12. Reboot the machine. Log in as user `student` and enter the password.

13. Click on the upper-left corner (the mouse-like Xubuntu start button) and start typing word `terminal`. Once you see **Terminal Emulator**, right-click it and select **Add to Desktop**. This would make easier to create Linux-like terminal windows and run command-lines.



3.3 Install Basic Software on Xubuntu

1. Set the root password to `Bitl1!` – same as for the user `student`:

```
sudo passwd   (enter Bitl1! password as student user)
    (type Bitl1! twice to set root's password)
```

2. Install all the software updates:

```
sudo apt-get update
sudo apt-get upgrade
```

3. Install Java JDK (prerequisite for Jenkins). First search all the "openjdk" related installations, then install the package `openjdk-11-jdk`. Finally, check if your Java has the right version 1.11.

```
sudo apt search openjdk
sudo apt-get install openjdk-11-jdk
java -version
```

4. Install C++ compiler (named `g++`) and also `make` utility:

   ```
   sudo apt-get install build-essential
   ```

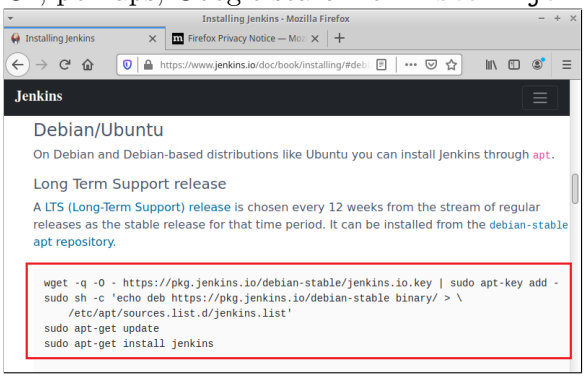
5. Install Git client:

   ```
   sudo apt-get install git
   ```

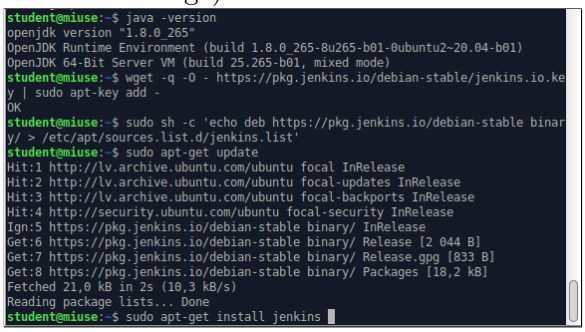
3.4 Setup of Jenkins

1. In the Xubuntu guest machine, click on the mouse-start button. Type in `Web Browser` to open Firefox-like browser.

2. Find the Jenkins installation commands for Debian/Ubuntu. Type this URL into the browser: `https://www.jenkins.io/doc/book/installing/#debianubuntu`.
   Or, perhaps, Google search for `install jenkins on ubuntu`:



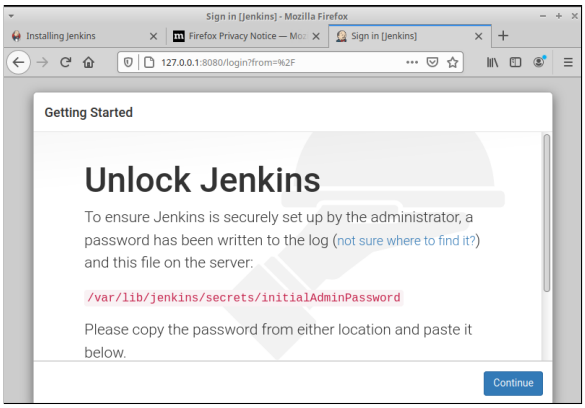
3. Copy-paste all the 4 commands into Xubuntu terminal (highlighted in red rectangle in the above image).



4. Register Jenkins as a system service that starts whenever Xubuntu is running:

   ```
   sudo systemctl start jenkins
   sudo systemctl status jenkins
   ```

5. In Xubuntu Web Browser enter `http://127.0.0.1:8080`.

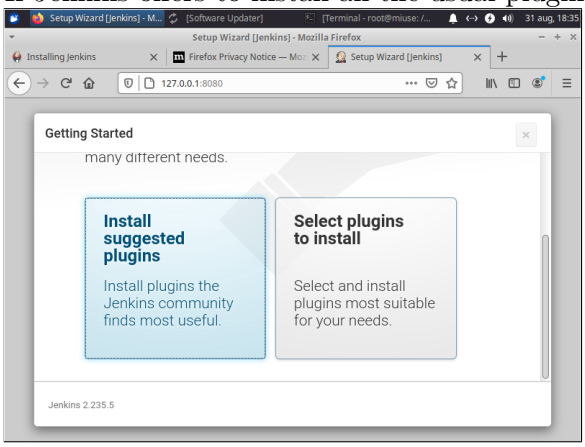


6. Change the user to `root` using "su - root" command and display the file containing the initial password of Jenkins.

```
student@miuse:~$ su - root
Password:
root@miuse:~# cd /var/lib/jenkins/secrets
root@miuse:/var/lib/jenkins/secrets# more initialAdminPassword
36259f6a02ff4f6f9e7051fcce825566
root@miuse:/var/lib/jenkins/secrets#
```

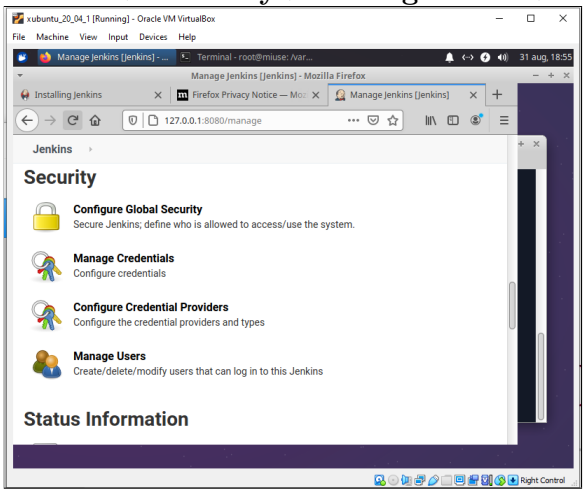
7. Copy-paste this one-time password into your browser, click **Continue**.

8. If Jenkins offers to install all the usual plugins, close the screen.

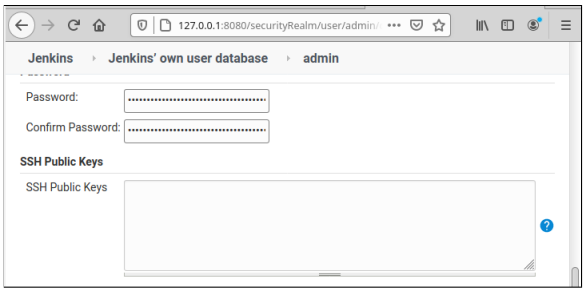


Note. Continuing with installing all the plugins might crash Jenkins instance.

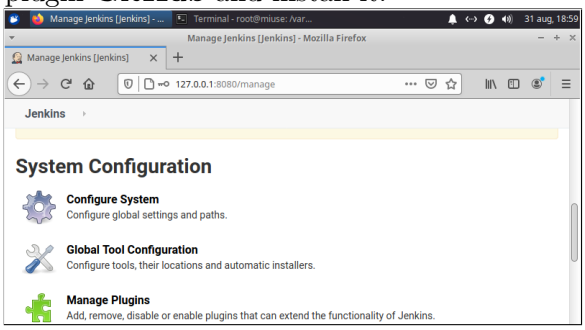
9. Reopen Browser, log into Jenkins again. In the Jenkins Web interface open **Manage Jenkins > Security > Manage Users > admin**.



10. Change the password to something easier, say `Bitl1!`



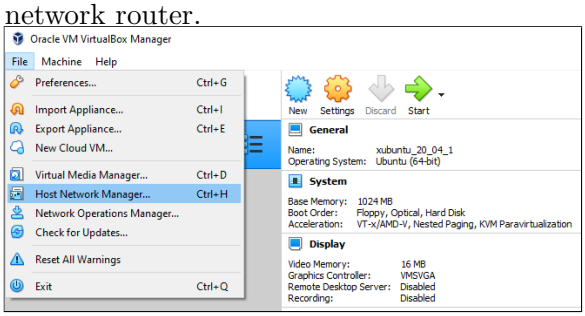
11. Navigate to **Manage Jenkins** > **Manage Plugins**. Open tab **Available**, search for the plugin **GitHub** and install it.



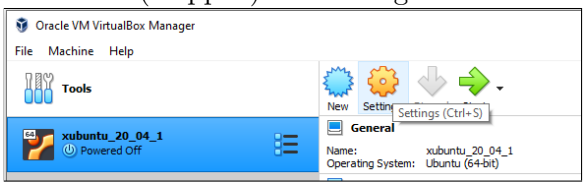
3.5 Fixing Host-Guest Networking

1. Shut down the Xubuntu guest.

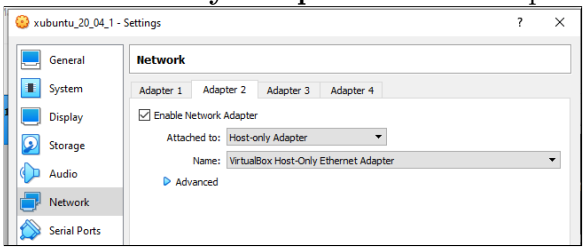
2. In VirtualBox application open **File** > **Host Network Manager** and inspect the host network router.



3. Select the (stopped) Xubuntu guest machine in VirtualBox and select **Settings**.

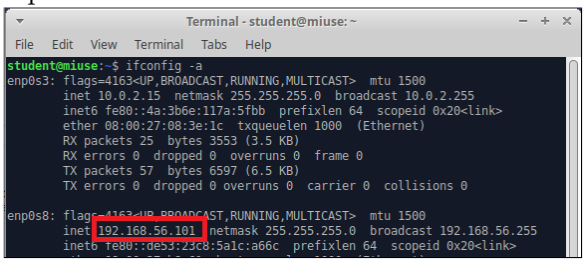


4. Open **Network** > **Adapter 2**. Select the **Enable Network Adapter** checkbox and select **Host-only Adapter** from the drop-down list.

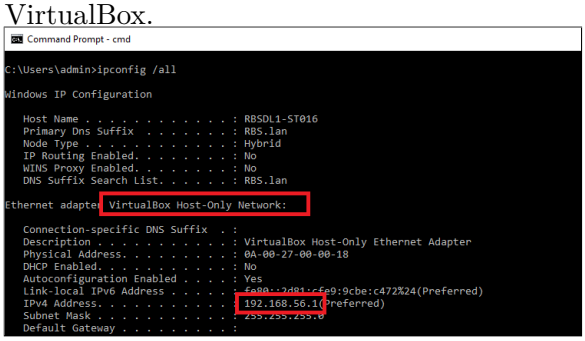


5. Run the Xubuntu guest machine (click on the green arrow).

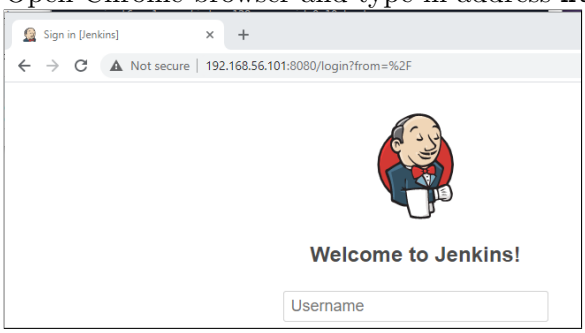
6. Open terminal window on Xubuntu and run command `ifconfig -a`



7. If you wish, open console on your host machine (such as Windows 10 or whatever). Type command `ipconfig /all`. You should be able to see a new network related to your VirtualBox.



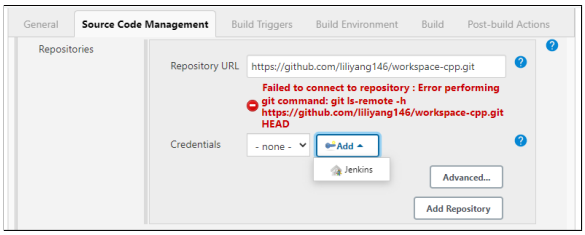
8. Use the Xubuntu address in the "Host-only network" to connect from your host machine. Open Chrome browser and type in address `http://192.168.56.101:8080`.



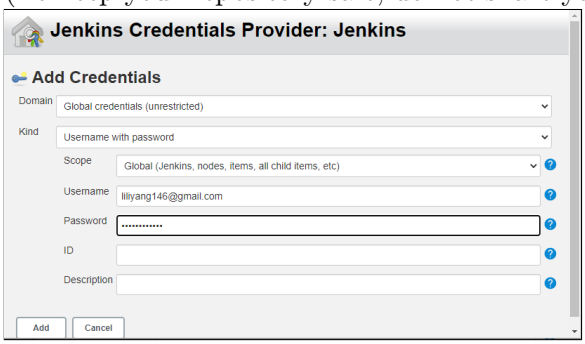
9. If Jenkins offers to make this URL to be "the Jenkins URL", agree by clicking **Save and Finish**. Log in using the credentials `admin` and `Bitl1!`.

3.6 Configure and Run a Jenkins Task

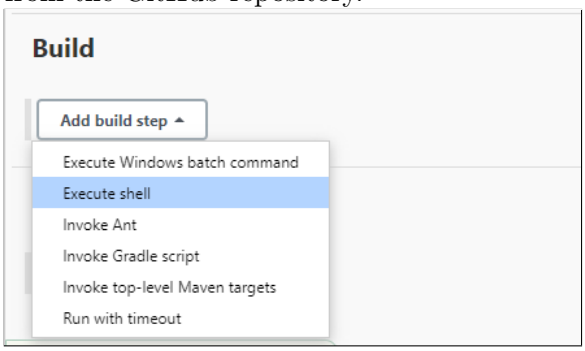
1. Create a private GitHub repository (`workspace-cpp` in our example, but you can name it however you want). Create a subdirectory `palindromes` containing some C++ sources and a makefile. You can copy the source code from this URL: `http://linen-tracer-682.appspot.com/data-structures-bin/palindromes.zip`.
2. Share the repository URL with your instructor (and add him as
3. Open `http://192.168.56.101:8080` and log in.
4. Enter some project name `test01-palindromes`, select **Freestyle project**.
5. Open tab **Source Code Management**, enter the repository URL of your GitHub repository.



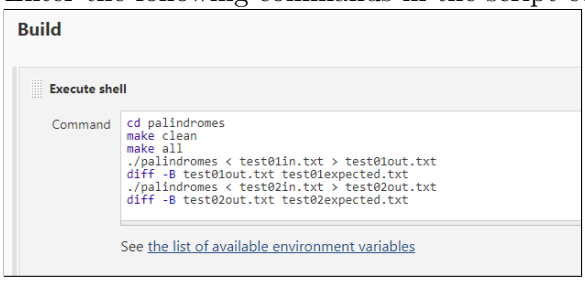
6. A red warning should be displayed. Add the credentials to log into your private repository. (To keep your repository safe, do not share your Xubuntu/Jenkins instance with others.)



7. Under subsection **Build** create a new build script to execute once the code is checked out from the GitHub repository.

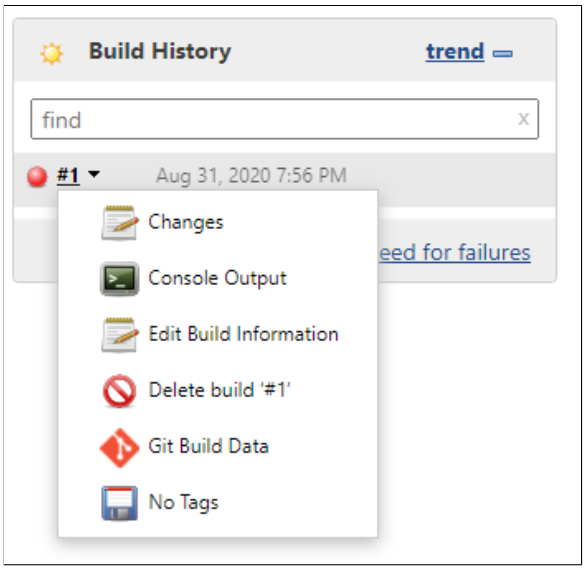


8. Enter the following commands in the script editor:



9. Click on button **Build Now** to execute your task.

10. You can inspect the **Console Output** if it failed.



11. The console commands with some output will be displayed on the browser screen:

12. If the console output is not sufficiently clear, you can locate the project in Jenkins workspace and run the build commands manually to find what is wrong.