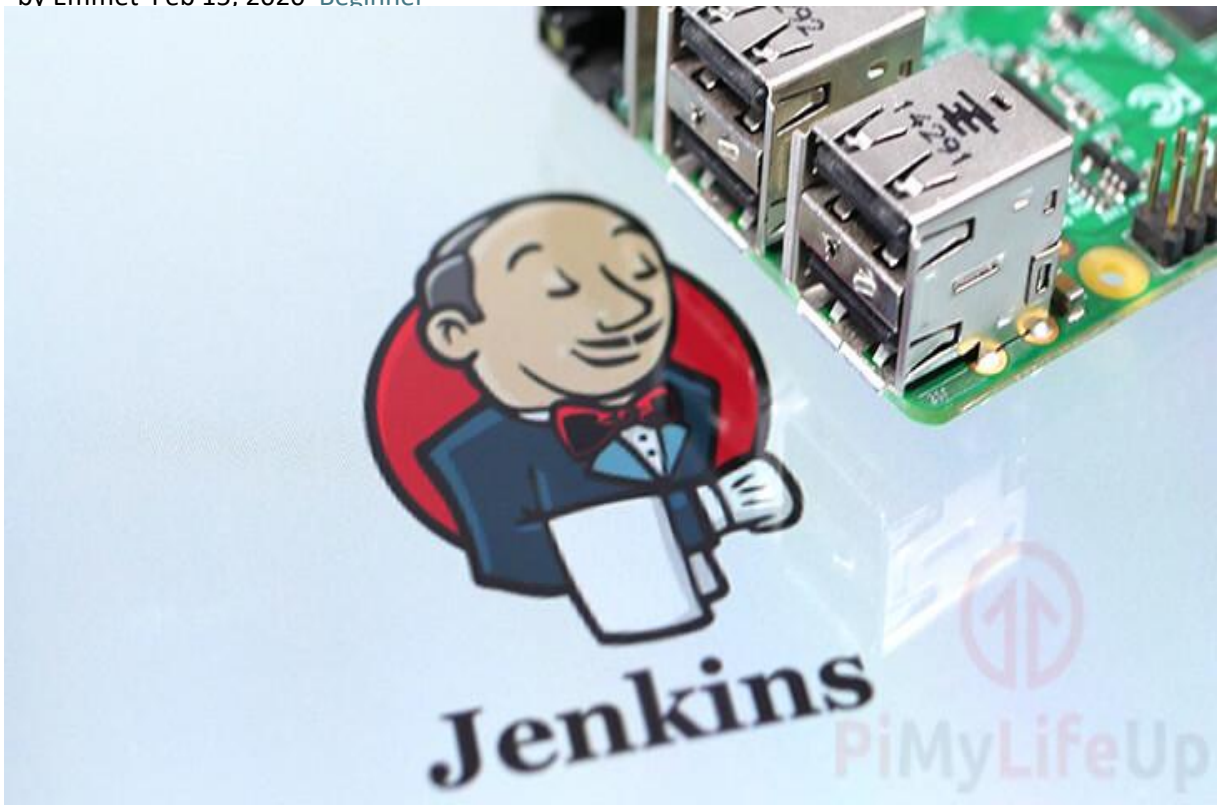
https://pimylifeup.com/jenkins-raspberry-pi/

Two ways to install…via these instructions or installing a Tomcat server and downloading the Jenkins War file.

Installing Jenkins on your Raspberry Pi

by Emmet Feb 15, 2020 Beginner



Jenkins is an open-source automation server designed to help automate tasks for software development.
The software includes functionality to handle both continuous integration and continuous delivery.
You can use Jenkins to compile your software automatically, run tests on it, and also “deliver” it to an end destination.

Raspberry Pi Chromium Kiosk

Preparing for Jenkins on the Raspberry Pi

1. To begin this guide, let's first upgrade all the pre-existing packages.

To do this, you need to run the following two commands.

```
sudo apt update
sudo apt upgrade
```

2. Now we need to install the latest version of Java to our Raspberry Pi.

We are required to install the Java runtime environment as Jenkins is built on top of the Java language.

Run the following command to install the OpenJDK 11 version of Java to your Pi.

```
sudo apt install openjdk-11-jre
```

3. Verify that you have successfully installed Java to your Raspberry Pi by running the command below.

```
java --version
```

From this command, you should see something like the following appear in the terminal.

```
openjdk 11.0.6 2020-01-14
OpenJDK Runtime Environment (build 11.0.6+10-post-Raspbian-1deb10u1)
OpenJDK Server VM (build 11.0.6+10-post-Raspbian-1deb10u1, mixed mode)
```

This response shows the version of OpenJDK that you are running and indicates that you have successfully installed Java.

Installing Jenkins to the Raspberry Pi

1. Before we continue, we need to add the Jenkins repository key to our Pi's key chain.

Adding the key will allow us to download Jenkins from their official package repository to our Raspberry Pi.

Run the following command to download and add the key.

```
wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add -
```

2. Our next step is to add the repository into our sources list.

We will do this by making a file called "jenkins.list" within the "/sources.list.d/" directory using the following command.

The package handler automatically reads the files located in the "/sources.list.d/" directory for links to repositories.

```
sudo nano /etc/apt/sources.list.d/jenkins.list
```

3. Add the following text to the file. This text defines the link to the Jenkins repository.

```
deb https://pkg.jenkins.io/debian binary/
```

Once added, save the file by pressing + , then , followed by .

4. With the new repository added, we need to go ahead and update the package list.

Update the list by running the following command.

```
sudo apt update
```

5. Now finally, we can install Jenkins by using the command below.

```
sudo apt install jenkins
```

This command will install Jenkins, create a user for it to run under, and set up the software's default configuration.

Unlocking Jenkins

To be able to set up Jenkins on your Raspberry Pi, you will need to unlock the setup screen.

1. If you don't know your Raspberry Pi's local IP address, you will need to obtain it by using the following command.

hostname -I

2. Next, we need to retrieve the initial admin password. Run the following command to get this password using the cat command. You will need the string returned by this in **step 4**.

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

From this command, you should end up with a result a bit like what we have below. This result is the password you need to unlock your Jenkins installation.

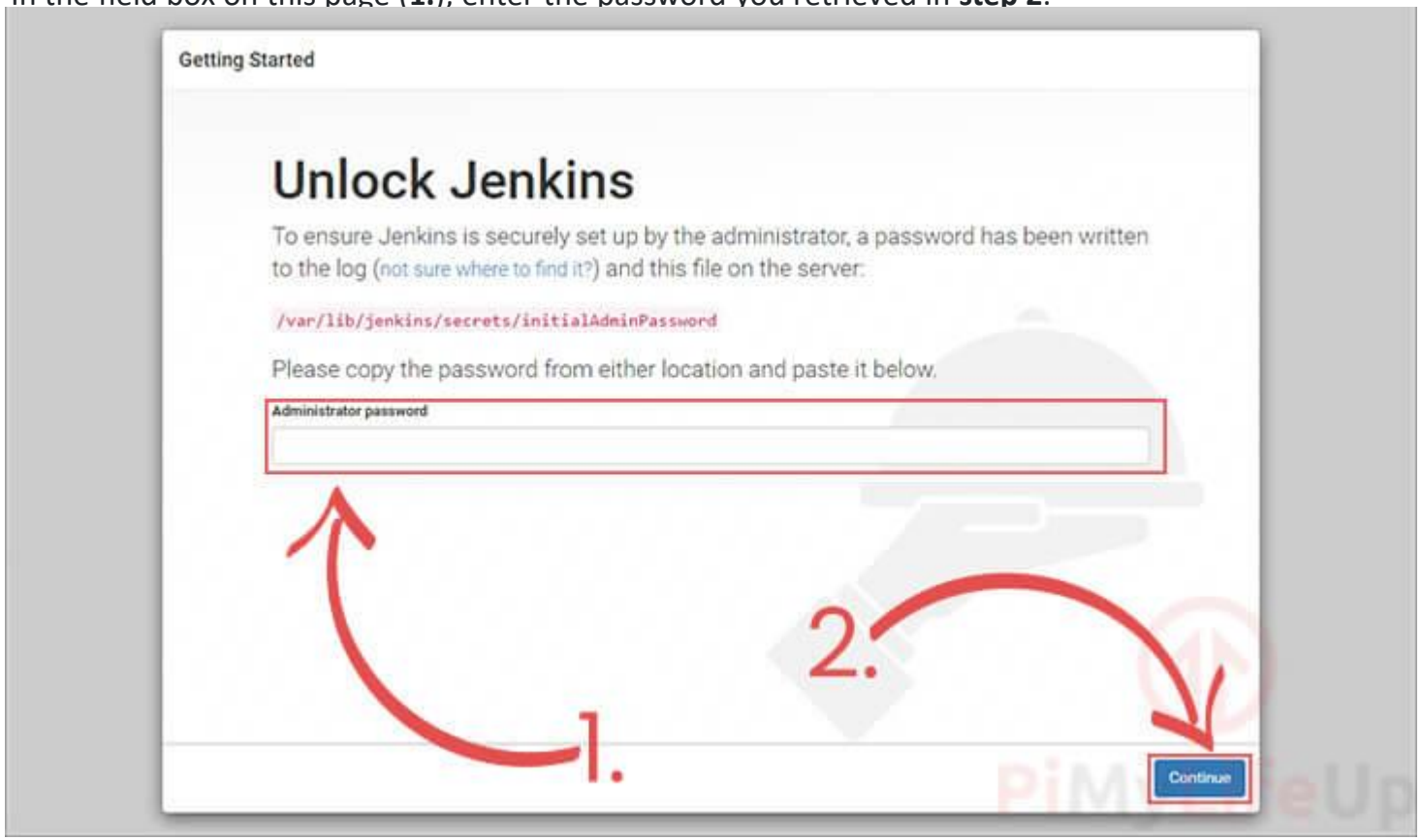
9df8c9c4cf694754a0047771a53b9fe8

3. We can now begin the process of setting up Jenkins by going to the following web address in your favorite web browser. Make sure that you replace "[RASPBERRYPIIPADDRESS]" with the IP that you retrieved in **step 1** of this section.

[RASPBERRYPIIPADDRESS]:8080

IE http://192.168.1.5:8080

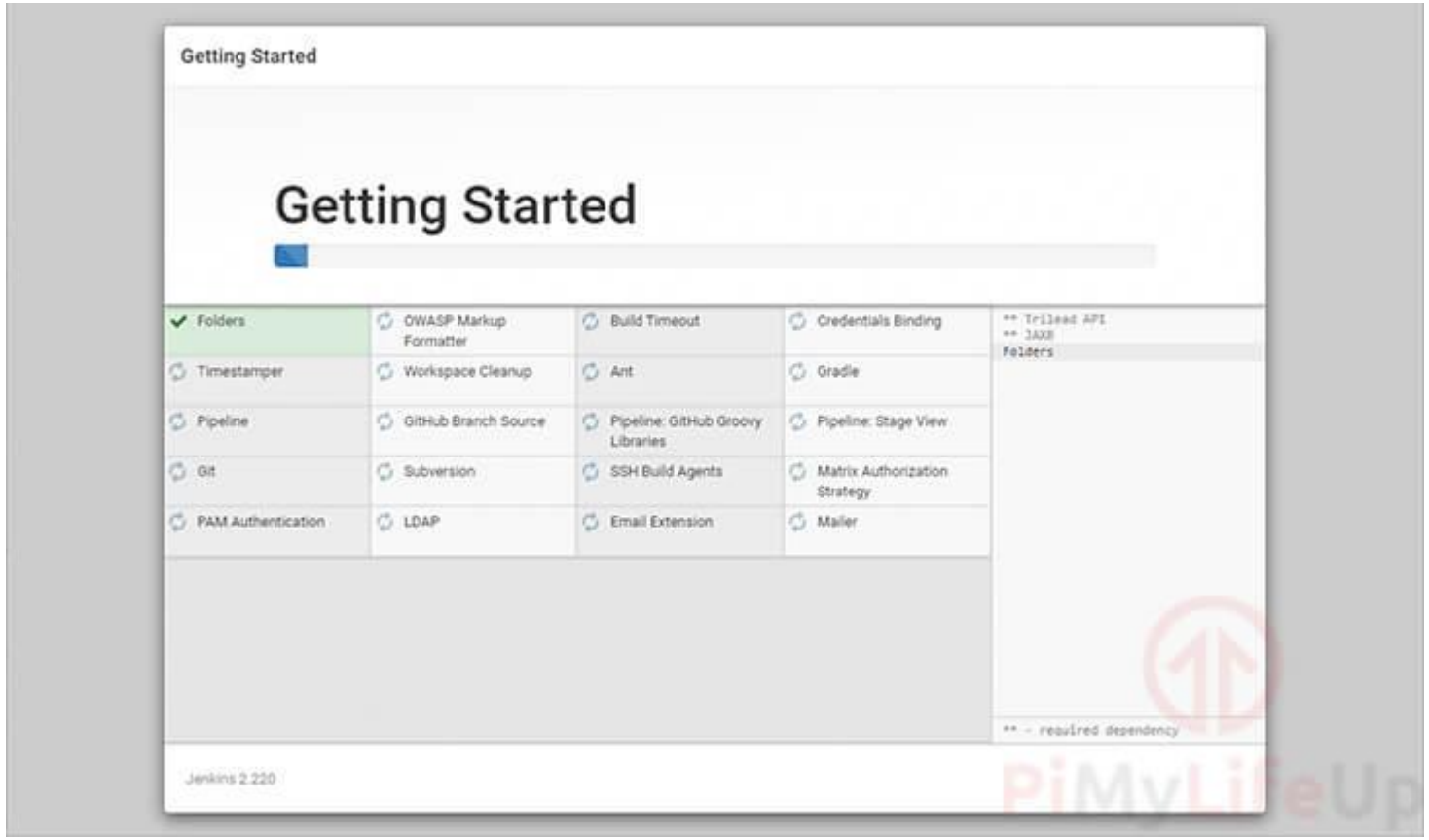
4. When you first go to the web address, you will be asked to unlock Jenkins.
In the field box on this page (**1.**), enter the password you retrieved in **step 2**.



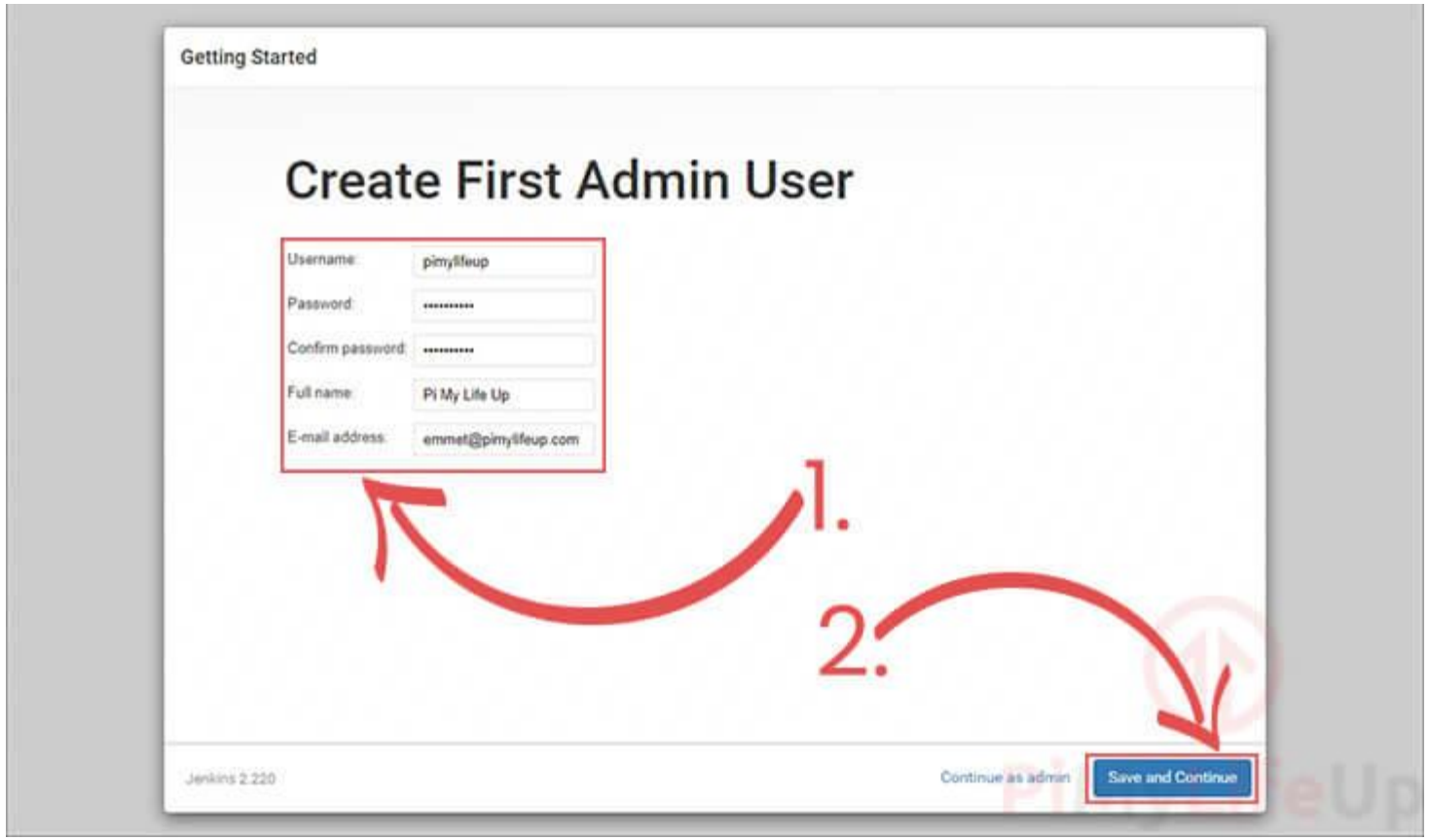
Setting Up Jenkins

1. The next screen will ask you if you want to install the suggested plugins or select them for yourself.
For this guide, we are just going to the "Install suggested plugins" button.

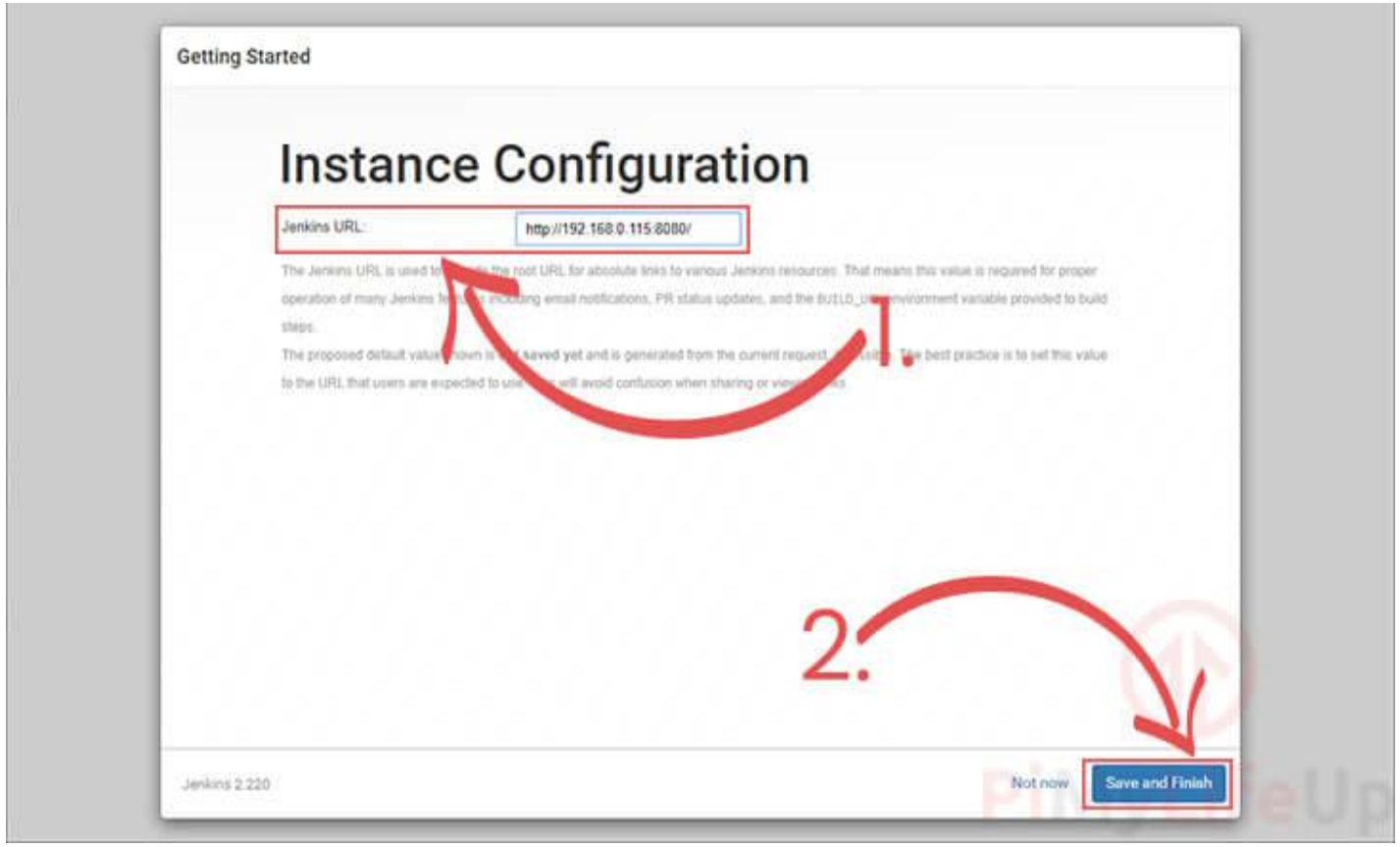
2. This process will take some time, so sit back and relax.



3. Next, you will be asked to create an admin user to interact with your Raspberry Pi Jenkins installation.
Enter the details for your user in the text boxes (**1.**).
Once you have entered all the details for your admin user, the "**Save and Continue**" button (**2.**).

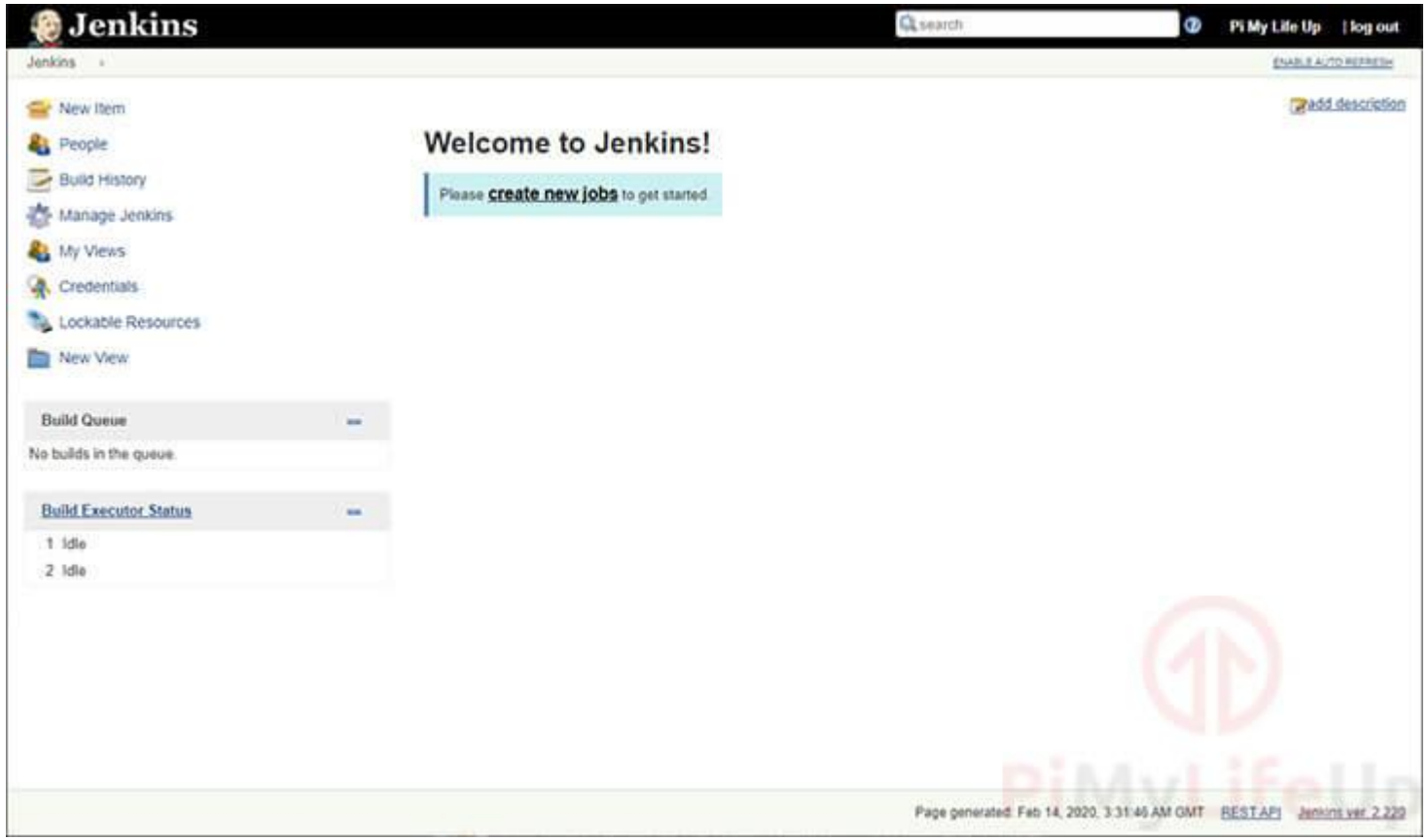


4. This screen will ask you to verify the URL it has generated for your Jenkins installation.
This URL should be your Raspberry Pi's IP address followed by the port number 8080.
Please note that you should have your Raspberry Pi set up with a static IP address, as a change in IP will break the installation.



5. Your Raspberry Pi Jenkins installation is now set up and ready to use.
All you need to do is the "**Start using Jenkins**" button to continue.

6. At this point, you should now have successfully installed Jenkins to your Raspberry Pi.



Controlling the Jenkins Service

If you ever want to control the Jenkins service, you can use sysmtectl

Retrieving the Status of the Jenkins Server

Use the following command to retrieve the current status of the Jenkins service.

```
sudo systemctl status jenkins
```

Starting the Jenkins Service

To start the Jenkins service, you can run the command below.

```
sudo systemctl start jenkins
```

Stopping the Jenkins Service

If you want to stop the Jenkins service, you can try using the command below.

```
sudo systemctl stop jenkins
```

Disabling the Jenkins Service

You can stop Jenkins from starting up at boot by disabling it with the following command.

```
sudo systemctl disable jenkins
```

Enabling the Jenkins Service

Likewise, you can also enable the Jenkins service, so that is starts at boot by using the command below.

```
sudo systemctl enable jenkins
```

Hopefully, you will now have an idea on how to install Jenkins to a Raspberry Pi. You should also now know how to start, stop, and disable the Jenkins service.

If you have run into any issues with getting Jenkins set up, then feel free to drop a comment below.

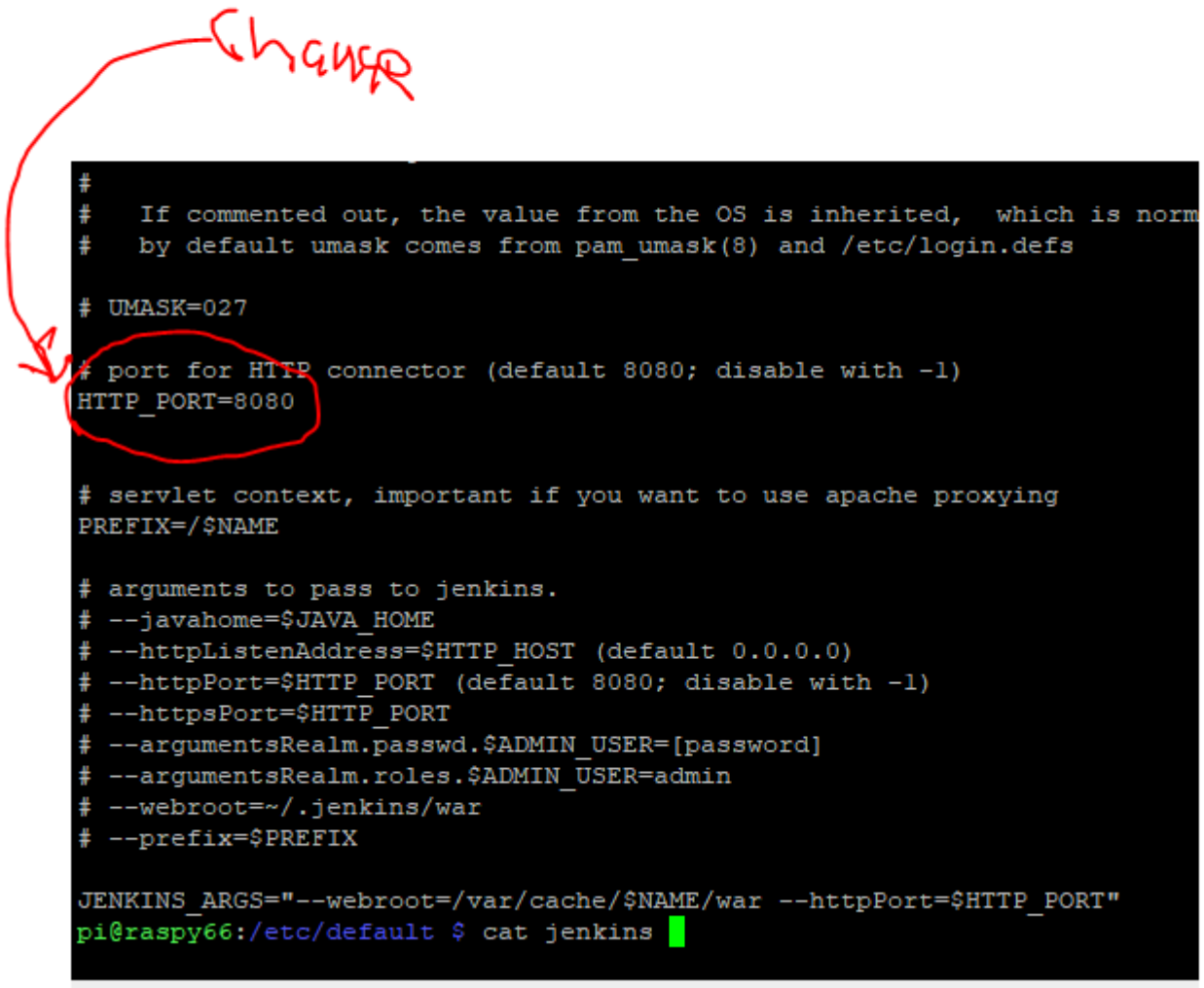
Jenkins Install Directory

/var/lib/**jenkins**.

```
pi@raspy66:/var/lib/jenkins $ ls -ltr
total 100
-rw-r--r--  1 jenkins jenkins    64 Oct  5 21:39 secret.key
-rw-r--r--  1 jenkins jenkins     0 Oct  5 21:39 secret.key.not-so-secret
drwxr-xr-x  2 jenkins jenkins  4096 Oct  5 21:39 nodes
-rw-r--r--  1 jenkins jenkins   156 Oct  5 21:39 hudson.model.UpdateCenter.xml
-rw-------  1 jenkins jenkins  1712 Oct  5 21:39 identity.key.enc
-rw-r--r--  1 jenkins jenkins   171 Oct  5 21:39 jenkins.telemetry.Correlator.xml
drwxr-xr-x  2 jenkins jenkins  4096 Oct  5 21:39 userContent
drwxr-xr-x  3 jenkins jenkins  4096 Oct  5 21:39 logs
-rw-r--r--  1 jenkins jenkins   907 Oct  5 21:39 nodeMonitors.xml
drwxr-xr-x 79 jenkins jenkins 12288 Oct  5 21:43 plugins
drwxr-xr-x  2 jenkins jenkins  4096 Oct  5 21:44 workflow-libs
-rw-r--r--  1 jenkins jenkins   370 Oct  5 21:44 hudson.plugins.git.GitTool.xml
-rw-r--r--  1 jenkins jenkins   475 Oct  5 21:44 com.cloudbees.hudson.plugins.folder.config.AbstractFolderConfiguration.xml
drwxr-xr-x  3 jenkins jenkins  4096 Oct  5 21:45 users
-rw-r--r--  1 jenkins jenkins   183 Oct  5 21:45 jenkins.model.JenkinsLocationConfiguration.xml
-rw-r--r--  1 jenkins jenkins     5 Oct  5 21:45 jenkins.install.UpgradeWizard.state
-rw-r--r--  1 jenkins jenkins     5 Oct  5 21:45 jenkins.install.InstallUtil.lastExecVersion
-rw-r--r--  1 jenkins jenkins  1642 Oct  5 21:45 config.xml
drwxr-xr-x  4 jenkins jenkins  4096 Oct  5 21:58 jobs
drwxr-xr-x  4 jenkins jenkins  4096 Oct  5 21:58 workspace
-rw-r--r--  1 jenkins jenkins  1300 Oct  6 06:39 hudson.plugins.emailext.ExtendedEmailPublisher.xml
drwx------  4 jenkins jenkins  4096 Oct  6 14:30 secrets
drwxr-xr-x  2 jenkins jenkins  4096 Oct  6 15:17 updates
-rw-r--r--  1 jenkins jenkins   130 Oct  6 18:59 queue.xml
pi@raspy66:/var/lib/jenkins $ pwd
```

To change default port from 8080

1. First, run this command to open **jenkins** configurations: sudo nano /etc/default/**jenkins**.
2. The only part you need to **change** is: **#port** for HTTP connector (default 8080; disable with -1) Http_port = 8080. ...
3. Finally, Restart **Jenkins** service by running this command: sudo service **jenkins** restart.



Setting up Jenkins on a Tomcat server installed.

Download a Tomcat file to unzip
https://tomcat.apache.org/download-90.cgi

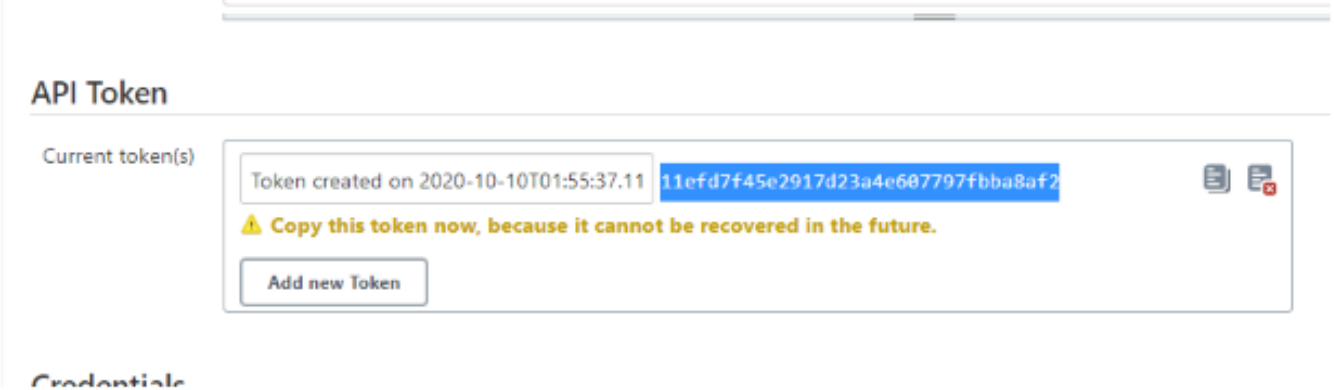
Start it as sudo su (Root)
cd /opt/tomcat8/apache-tomcat-version/bin
./startup.sh
Once done, log in via port 8080,.
Do normal setup as per instructions above.

Setting up Jenkins on Tomcat installed as a service on Raspberry PI

1.sudo apt-get install tomcat8
2.Sudo service tomcat8 stop (stop tomcat to access port 8080)
3.Run the steps above to install Jenkins as a service.
sudo apt-get update
.
.
.steps above wget…
.
.
sudo apt install jenkins

4.Once jenkins is installed, change to port # to 8084. (directions above)
1.Stop/start jenkins service
sudo service jenkins stop && sudo service jenkins start
http://192.168.1.xx:8084
6.sudo service tomcat8 start (restart tomcat)

Running jenkins from command line
https://www.jenkins.io/doc/book/managing/cli/



Token created
11efd7f45e2917d23a4e607797fbba8af2

From <http://192.168.1.97:8080/jenkins/me/configure>

Get port number
curl -Lv https:///login 2>&1 | grep -j 'x-ssh-endpoint'

From <https://www.jenkins.io/doc/book/managing/cli/>

Need to export the token in a environment variable.

```
pi@jenkinspi:~ $ export JENKINS_USER_ID=pi
pi@jenkinspi:~ $ export JENKINS_API_TOKEN=11efd7f45e2917d23a4e607797fbba8af2
pi@jenkinspi:~ $ java -jar cli-2.249.1.jar -s http://192.168.1.97:8080/jenkins/ build TestDir
pi@jenkinspi:~ $ java -jar cli-2.249.1.jar -s http://192.168.1.97:8080/jenkins/ build TestDir
pi@jenkinspi:~ $ java -jar cli-2.249.1.jar -s http://192.168.1.97:8080/jenkins/ build TestDir
pi@jenkinspi:~ $
```

********** finding the cli jar file **********

Found the command CLI jar file in following directory - for warfile installation found it where the war file was expanded.

IE

```
root@jenkinspi:/opt/tomcat9/apache-tomcat-9.0.38/webapps/jenkins/WEB-INF/lib# ls -ltr *cli*
-rw-r----- 1 root root  321537 Sep  9 14:06 commons-httpclient-3.1-jenkins-1.jar
-rw-r----- 1 root root   19162 Sep  9 14:06 stapler-adjunct-zeroclipboard-1.3.5-1.jar
-rw-r----- 1 root root    4414 Sep  9 14:06 robust-http-client-1.2.jar
-rw-r----- 1 root root   12838 Sep  9 14:11 ssh-cli-auth-1.8.jar
-rw-r----- 1 root root 3141038 Sep  9 15:52 cli-2.249.1.jar
root@jenkinspi:/opt/tomcat9/apache-tomcat-9.0.38/webapps/jenkins/WEB-INF/lib#
```

cp cli-2.249.1.jar /home/pi directory (or your home directory)

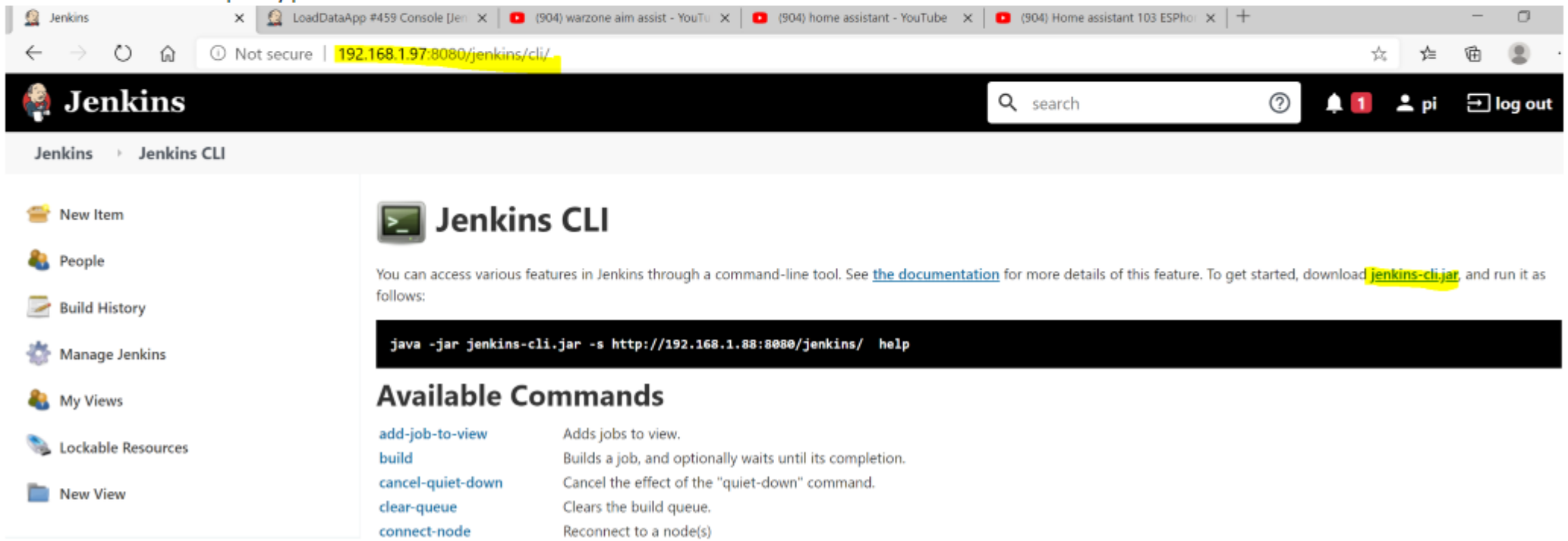
chown pi:pi cli-2.249.1.jar

chmod 776 cli-2.249.1.jar

Alternatively you can download the file from Jenkins own website from your Jenkins url under 'manage jenkins'->'jenkins cli' links.

Then hitting the highlighted below.

This works with both raspberry pi installer and the war file installation.



Information on cli.
http://192.168.1.97:8080/jenkins/cli/

To run a build IE.

- java -jar jenkins-cli.jar -s http://192.168.1.88:8080 build [jobname]
- java -jar jenkins-cli.jar -s jenkins_url console [jobname] (outputs latest build console log)

To print console output

java -jar jenkins-cli.jar -s http://192.168.1.88:8080/jenkins/ console JOB [BUILD] [-f] [-n N]
Produces the console output of a specific build to stdout, as if you are doing 'cat build.log'

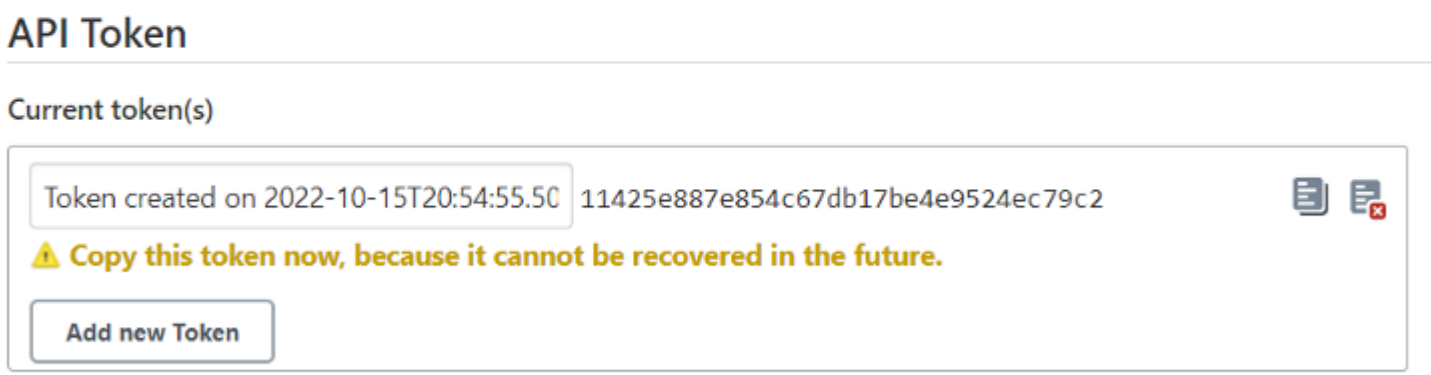
JOB : Name of the job
BUILD : Build number or permalink to point to the build. Defaults to the last build (default: lastBuild)
-f : If the build is in progress, stay around and append console output as it comes, like 'tail -f' (default: false)
-n N : Display the last N lines (default: -1)

From <http://192.168.1.97:8080/jenkins/cli/command/console>

Listing output of console for specific job

```
pi@jenkinspi:~ $ java -jar cli-2.249.1.jar -s http://192.168.1.97:8080/jenkins/ console TestDir 670
Started from command line by ha:////4CBDrFvhqJQfdIGOkAPs8BOjJITx6dvAioF6LWFC7oc4AAAAlB+LCAAAAAAAAAP9b85aBtbiIQTGjNKU4P08vOT+vOD8nVc83PyUlx6
Running as SYSTEM
Building in workspace /root/.jenkins/workspace/TestDir
[TestDir] $ /bin/sh -xe /opt/tomcat9/apache-tomcat-9.0.38/temp/jenkins1273846356802702904 5.sh
+ cd /home/pi
+ ls -ltr
total 142196
-rwxrwxrw- 1 pi   pi   142445568 Oct  9 03:39 ejdk-8u211-linux-arm-sflt.tar
drwxr-xr-x 6 pi   pi        4096 Oct  9 03:45 ejdk1.8.0_211
-rwxrwxrw- 1 pi   pi     3141038 Oct 10 02:22 cli-2.249.1.jar
-rw-r----- 1 root root     12838 Oct 10 02:22 ssh-cli-auth-1.8.jar
[TestDir] $ /bin/sh -xe /opt/tomcat9/apache-tomcat-9.0.38/temp/jenkins1486919166648044087 8.sh
+ cd /var
+ ls -ltr
total 102436
drwxrwsr-x  2 root staff      4096 Feb  9  2020 local
drwxr-xr-x  2 root root       4096 Feb 13  2020 opt
drwxrwsr-x  2 root mail       4096 Feb 13  2020 mail
lrwxrwxrwx  1 root root          4 Feb 13  2020 run -> /run
lrwxrwxrwx  1 root root          9 Feb 13  2020 lock -> /run/lock
```

Keys for testing (to be deleted)
Jenkins key for jenkinspi
11efd7f45e2917d23a4e607797fbba8af2
Jenkins Key for raspy66
111492e10a068e4f9c6e320690a039448c



ken taylor admin
11425e887e854c67db17be4e9524ec79c2