TABLE OF CONTENTS

TOPIC PAGE

Installation of Java8…………………………………………………………………….. 2

Installation of Apache Maven………………………………………………………. 3

Installation of NodeJS using NPM………………………………………………… 4

Installation of Docker…………………………………………………………………… 5

Installation of Angular CLI…………………………………………………………….. 7

Installation of Jenkins……………………………………………………………………. 8

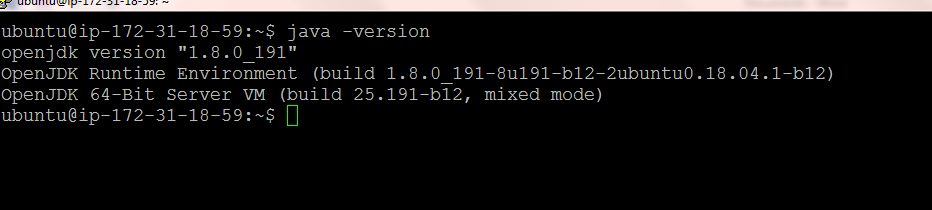
Installation of SonarQube……………………………………………………………… 9

Installation of Nexus Repository Manager…………………………………….. 13

References……………………………………………………………………………………..

How to Install Java8.

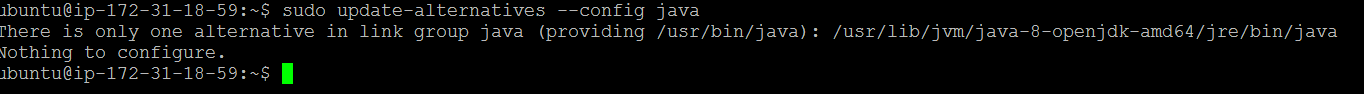
* In order to be able to install packages on your Ubuntu system, you must be logged in as a user with sudo privileges.
* To install this version, first update the package index:
  + sudo apt update
  + sudo apt install openjdk-8-jdk
* Verify that this is installed with
  + java –version



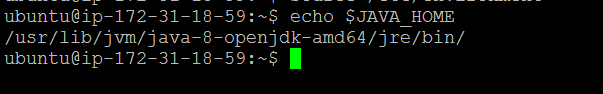
Setting the JAVA\_HOME Environment Variable

Many programs written using Java use the JAVA\_HOME environment variable to determine the Java installation location.

* To set this environment variable, first determine where Java is installed. Use the update-alternatives command.
* This command shows installation of Java along with its installation path
  + sudo update-alternatives --config java

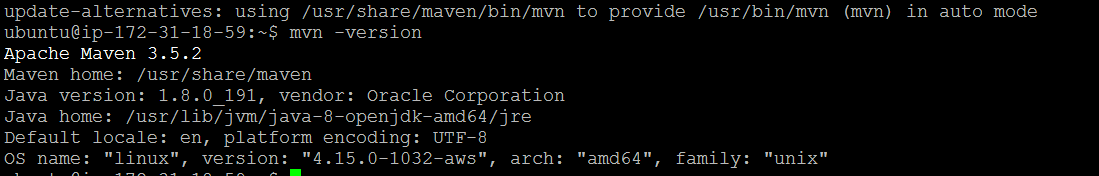


* Copy the path from your preferred installation. Then open /etc/environment using vi or your favorite text editor:
  + sudo vi /etc/environment
* At the end of this file, add the following line, making sure to replace the highlighted path with your own copied path
  + JAVA\_HOME="/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/"
* Modifying this file will set the JAVA\_HOME path for all users on your system.
* Save the file and exit the editor.
* Now reload this file to apply the changes to your current session:
  + source /etc/environment
* Verify that the environment variable is set:
  + echo $JAVA\_HOME
* You'll see the path you just set:



## [Installing Apache Maven on Ubuntu with Apt](https://linuxize.com/post/how-to-install-apache-maven-on-ubuntu-18-04/#installing-apache-maven-on-ubuntu-with-apt)

* Start by updating the package index:
  + sudo apt update
* Next, install Maven by typing the following command:
  + sudo apt install maven
* Verify the installation by running the mvn -version command:
  + mvn –version
* The output should look something like this:



## [Install Node.js & NPM from the NodeSource repository](https://linuxize.com/post/how-to-install-node-js-on-ubuntu-18.04/#install-node-js-from-the-nodesource-repository)

* Enable the NodeSource repository with the following curl command.
  + curl -sL https://deb.nodesource.com/setup\_10.x | sudo bash -
* Once the NodeSource repository is enabled, install Node.js and npm by typing:
  + sudo apt-get install -y nodejs
* Verify that the Node.js and npm were successfully installed is by printing their versions:
  + node –-version

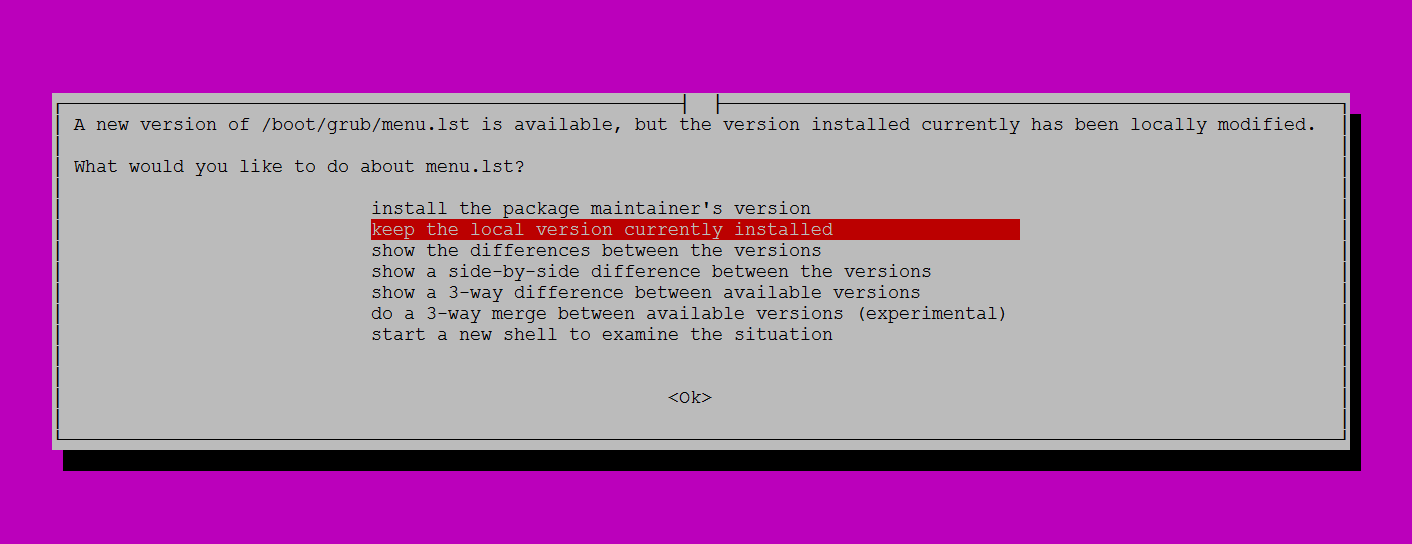


* + npm –-version

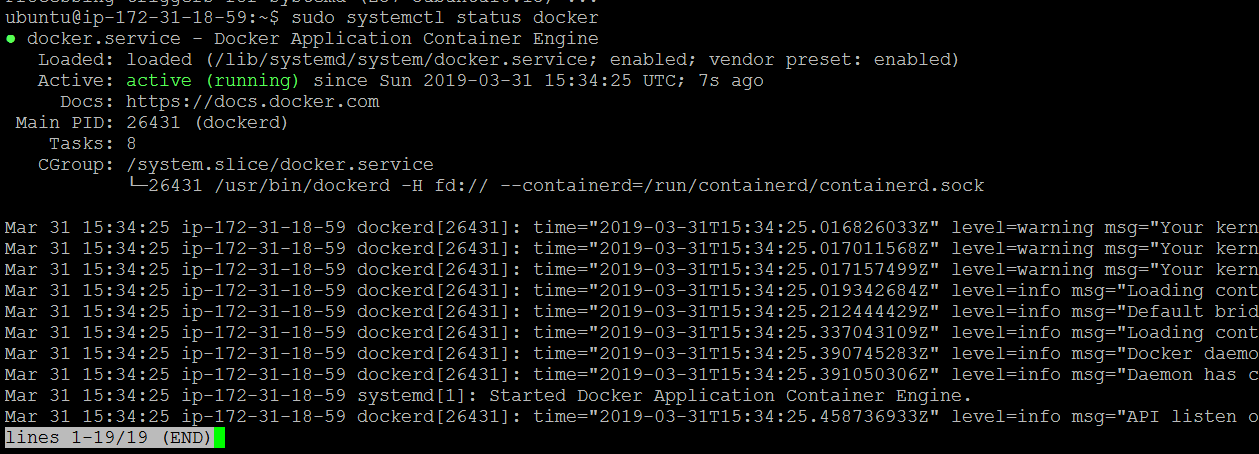


## [Install Docker on Ubuntu](https://linuxize.com/post/how-to-install-and-use-docker-on-ubuntu-18-04/#install-docker-on-ubuntu)

* First, update the packages index and upgrade your system with the following commands:
  + sudo apt update && sudo apt upgrade
* If you encounter any prompt like below, please click ok & proceed.



* Next, Install the dependencies necessary to enable a new repository over HTTPS:
  + sudo apt install apt-transport-https ca-certificates curl software-properties-common
* Import the repository’s GPG key using the following curl command.
  + curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –
* Add the Docker APT repository to your system’s software repository list by typing:
  + sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb\_release -cs) stable"
* Now that the Docker repository is enabled, update the apt package list and install the latest version of Docker CE (Community Edition) with:
  + sudo apt update && sudo apt install docker-ce
* Once the installation is completed the Docker service will start automatically. You can verify it by typing:
  + sudo systemctl status docker
* The output will look something like this:



* Check the Docker version by typing:
  + docker –v



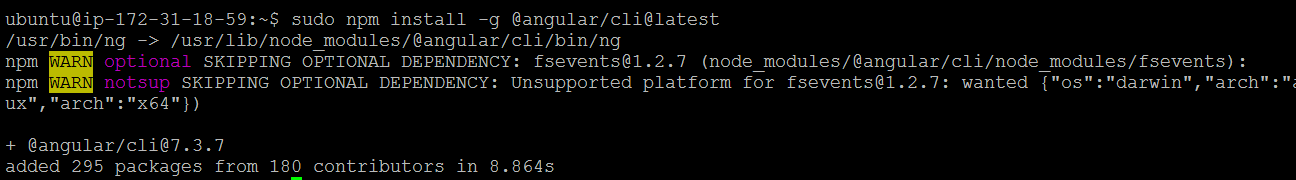
## [Executing the Docker Command Without Sudo](https://linuxize.com/post/how-to-install-and-use-docker-on-ubuntu-18-04/#executing-the-docker-command-without-sudo)

By default managing, Docker requires administrator privileges. If you want to run Docker commands as a non-root user without prepending sudo you need to add your user to the docker group which is created during the installation of the Docker CE package. You can do that by typing:

* + sudo usermod –aG docker $USER

## [Install Angualr CLI on Ubuntu](https://linuxize.com/post/how-to-install-and-use-docker-on-ubuntu-18-04/#install-docker-on-ubuntu)

* Install globally using below command.
  + sudo npm install -g @angular/cli@latest

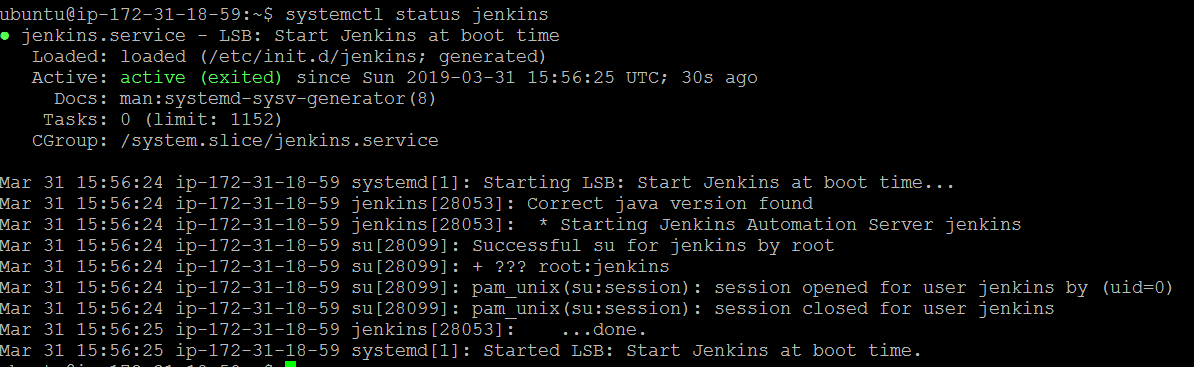


* Check the angular cli version by typing:
  + ng –version



# Install Jenkins on Ubuntu 18.04

* To install Jenkins on your Ubuntu system, follow these steps:
  + sudo apt update
  + wget -q -O - https://pkg.jenkins.io/debian/jenkins.io.key | sudo apt-key add –
  + sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'
  + sudo apt update
  + sudo apt install Jenkins
* Jenkins service will automatically start after the installation process is complete. You can verify it by printing the service status:
  + systemctl status Jenkins
* You should see something similar to this:



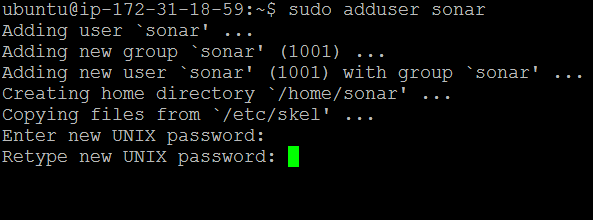
## [Adjusting Firewall](https://linuxize.com/post/how-to-install-jenkins-on-ubuntu-18-04/#adjusting-firewall)

* If you are installing Jenkins on a remote Ubuntu server that is protected by a firewall you’ll need to open port 8080. Make sure you mention port 8080 in inbound rules of azure VM Networking settings.
* To set up your new Jenkins installation, open your browser, type your domain or IP address followed by port 8080, http://your\_ip\_or\_domain:8080

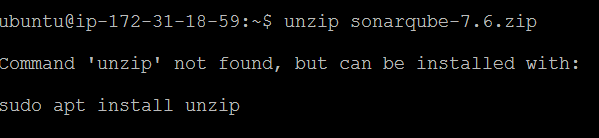
## Install and Configure SonarQube

Before starting, you will need to update your system with the latest version by running the below commands.

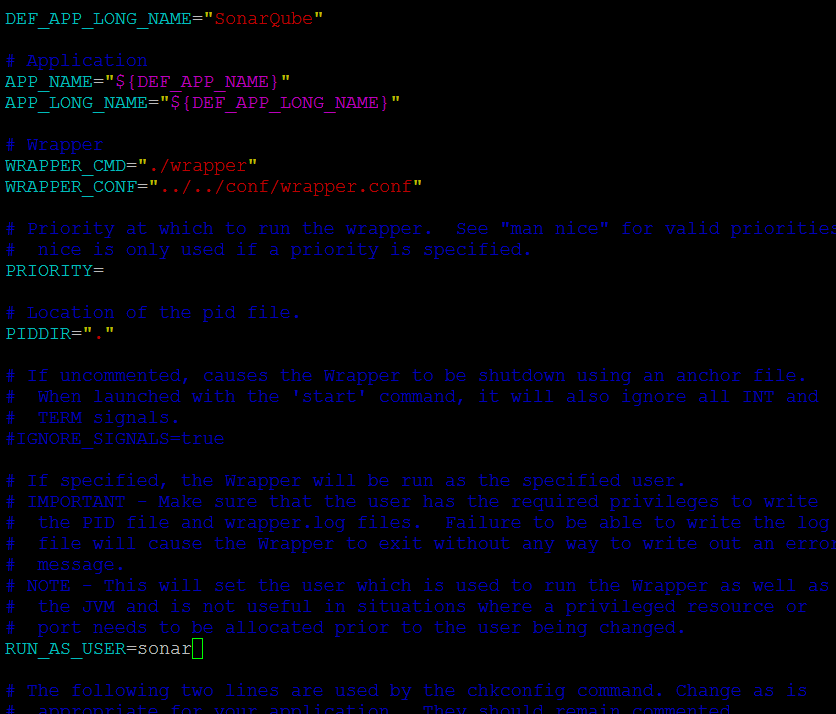
* + sudo apt update
  + sudo apt upgrade
* First, create a user for SonarQube with the following command:
  + sudo adduser sonar



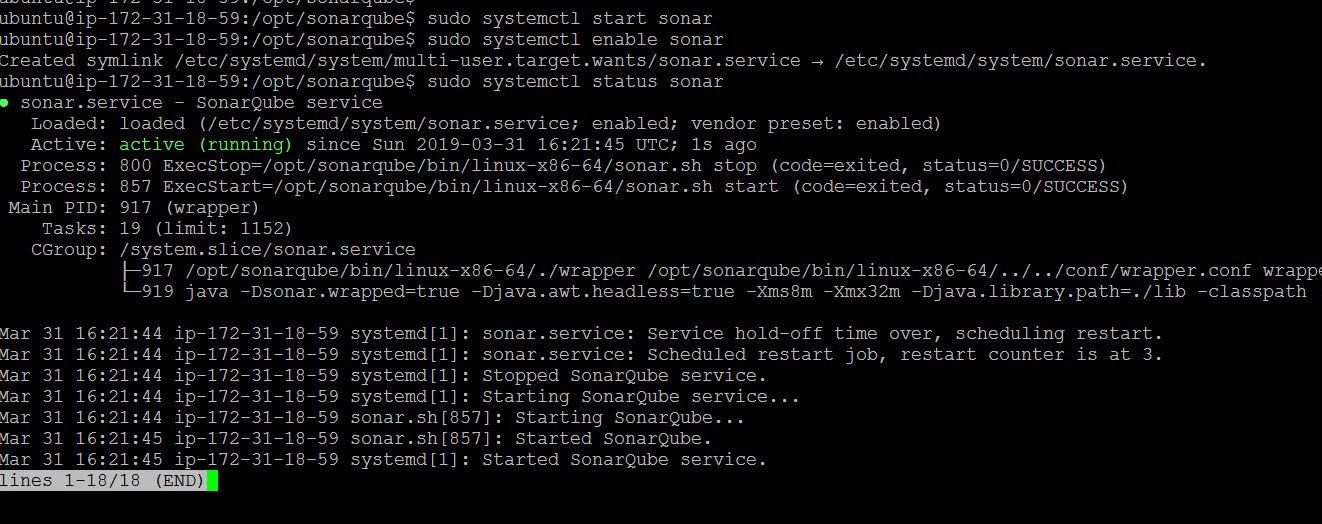
* Enter your desired password & confirm the same and accept the default values if it prompts.
* Next, download the latest version of SonarQube with the following command:
  + wget <https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-7.6.zip>
* Once the download is completed, unzip the downloaded file with the following command:
  + unzip sonarqube-7.6.zip
* If the below error comes, install unzip by using **sudo apt install unzip**



* Next, copy the extracted directory to the /opt with the following command:
  + sudo cp -r sonarqube-7.6 /opt/sonarqube
* Next, give ownership to the sonar user with the following command:
  + sudo chown -R sonar:sonar /opt/sonarqube
* Next, you will need to configure SonarQube to run as a sonar user. You can do this with the following command:
  + sudo vi /opt/sonarqube/bin/linux-x86-64/sonar.sh
* Make the following changes:
  + RUN\_AS\_USER=sonar



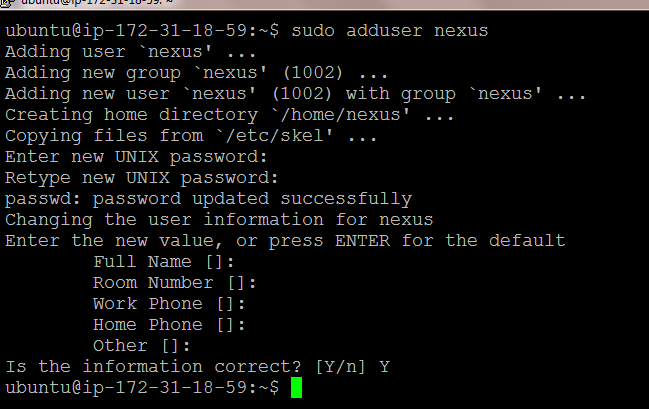
* Next, you will need to create a systemd service file to manage SonarQube service. You can do this with the following command:
  + sudo vi /etc/systemd/system/sonar.service
* Add the following lines:
* [Unit]
* Description=SonarQube service
* After=syslog.target network.target
* [Service]
* Type=forking
* ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start
* ExecStop=/opt/sonarqube/bin/linux-x86-64/sonar.sh stop
* User=sonar
* Group=sonar
* Restart=always
* [Install]
* WantedBy=multi-user.target
* Save and close the file, when you are finished. Then, start SonarQube service and enable it to start on boot time with the following command:
  + sudo systemctl start sonar
  + sudo systemctl enable sonar
  + sudo systemctl status sonar



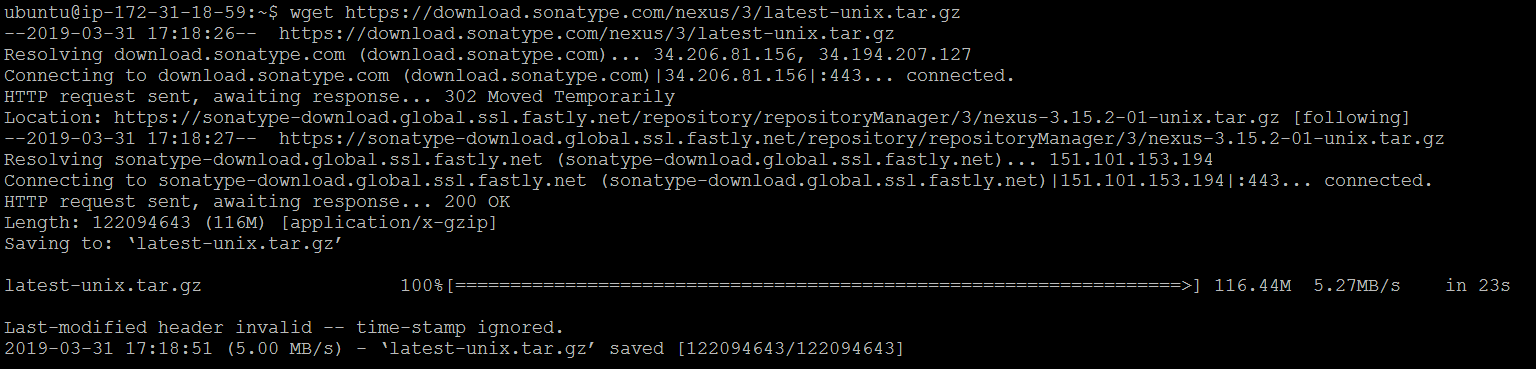
* SonarQube is now installed and configured, open your web browser and type the URL <http://your_ip_or_domain:9000>

## Install and Configure Nexus 3

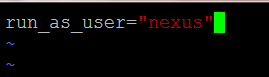
* To install Nexus Repository Manager in your Ubuntu system, follow the below steps.
  + sudo apt update
* First, create a user for Nexus with the following command:
  + sudo adduser nexus
* Enter your desired password & confirm the same and accept the default values if it prompts.



* Next, download the latest version of Nexus with the following command:
  + wget <https://download.sonatype.com/nexus/3/latest-unix.tar.gz>



* Unpack Nexus using tar –xvzf latest-unix.tar.gz
* Next, copy the extracted directory to the /opt with the following command:
  + sudo cp -r nexus-3.15.2-01 /opt/nexus
  + sudo cp -r sonatype-work /opt/
* Next, give ownership to the sonar user with the following command:
  + sudo chown -R nexus:nexus /opt/nexus
  + sudo chown -R nexus:nexus /opt/sonatype-work
* Next, you will need to configure Nexus to run as a nexus user. You can do this with the following command:
  + sudo vi /opt/nexus/bin/nexus.rc
* Make the following changes:



* Add this line in /opt/nexus/bin/nexus file
  + NEXUS\_HOME="/opt/nexus"
* Make the nexus configuration executable
  + sudo chmod a+x /opt/nexus/bin/nexus
* Next, you will need to create a systemd service file to manage Nexus service. You can do this with the following command:
  + sudo vi /etc/systemd/system/nexus.service
* Add the following lines:
* [Unit]
* Description=nexus service
* After=network.target
* [Service]
* Type=forking
* LimitNOFILE=65536
* ExecStart=/opt/nexus/bin/nexus start
* ExecStop=/opt/nexus/bin/nexus stop
* User=nexus
* Restart=on-abort
* [Install]
* WantedBy=multi-user.target
* Activate the service with the following commands:
  + sudo systemctl daemon-reload
  + sudo systemctl enable nexus.service
  + sudo systemctl start nexus.service
  + sudo systemctl status nexus.service
* Nexus is now installed and configured, open your web browser and type the URL <http://your_ip_or_domain:8081/>

# Install Azure CLI

* Get packages needed for the install process:
  + sudo apt-get update
  + sudo apt-get install curl apt-transport-https lsb-release gpg
* Download and install the Microsoft signing key:
  + curl -sL https://packages.microsoft.com/keys/microsoft.asc | \

gpg --dearmor | \

sudo tee /etc/apt/trusted.gpg.d/microsoft.asc.gpg > /dev/null

* Add the Azure CLI software repository:
  + AZ\_REPO=$(lsb\_release -cs)

echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ $AZ\_REPO main" | \

sudo tee /etc/apt/sources.list.d/azure-cli.list

* Update repository information and install the azure-cli package:
  + sudo apt-get update
  + sudo apt-get install azure-cli
* Run the Azure CLI with the az command. To sign in, use the [az login](https://docs.microsoft.com/en-us/cli/azure/reference-index#az-login) command.
  + az login
* If the CLI can open your default browser, it will do so and load a sign-in page.Otherwise, you need to open a browser page and follow the instructions on the command line to enter an authorization code after navigating to <https://aka.ms/devicelogin> in your browser.
* Sign in with your account credentials in the browser.

## References:

* <https://www.digitalocean.com/community/tutorials/how-to-install-java-with-apt-on-ubuntu-18-04>
* <https://linuxize.com/post/how-to-install-apache-maven-on-ubuntu-18-04/>
* <https://linuxize.com/post/how-to-install-node-js-on-ubuntu-18.04/>
* <https://linuxize.com/post/how-to-install-and-use-docker-on-ubuntu-18-04/>
* <https://www.npmjs.com/package/@angular/cli>
* <https://linuxize.com/post/how-to-install-jenkins-on-ubuntu-18-04/#adjusting-firewall>
* <https://www.howtoforge.com/how-to-install-sonarqube-on-ubuntu-1804/>
* <https://help.sonatype.com/learning/repository-manager-3/first-time-installation-and-setup/lesson-1%3A--installing-and-starting-nexus-repository-manager>
* <https://www.build-business-websites.co.uk/install-nexus-on-ubuntu-16-04/>