# Jenkins Installation on Linux (Ubuntu)

* Login to AWS Console
* Create EC2 Instance with Ubuntu AMI
* Login to EC2 Instance from Git Bash or any other terminal
* Execute below Commands
* This is the Debian package repository of Jenkins to automate installation and upgrade. To use this repository, first add the key to your system:

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc > /dev/null

* Then add a Jenkins apt repository entry:

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/ | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null

* Update your local package index, then finally install Jenkins:

sudo apt-get update -y

sudo apt-get install fontconfig openjdk-11-jre -y

* Install Jenkins

sudo apt-get install jenkins -y

* Start the Jenkins

sudo service jenkins start

* Install Git and Maven

sudo apt-get install git maven -y

* Add port no 8080 in Security group 🡪 Inbound Rules
* Open Browser and type
  + <publicip>:8080
* Jenkins will be opened in a browser
  + Unlock the Jenkins by following the instructions

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

* + - copy the password and paste
  + Install suggested Plugins
  + Create admin user
  + Then your Jenkins is ready

# Tomcat Configuration

1. Tomcat configuration in Linux System

* Create EC2 Instance (Red Hat EL OS) (This is another EC2 Instance)
* Instance Type t2. micro
* Login to EC2 from Git Bash/Terminal

**sudo su**

**yum install java-1.8\* wget -y**

**cd /opt**

**wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.76/bin/apache-tomcat-9.0.76.tar.gz**

* if you are not able to download the above package, Goto
  + [Apache Tomcat® - Welcome!](http://tomcat.apache.org/) and check for the latest version

**tar -xvzf apache-tomcat-9.0.76.tar.gz**

**chmod +x /opt/apache-tomcat-9.0.76/bin/startup.sh**

**chmod +x /opt/apache-tomcat-9.0.76/bin/shutdown.sh**

* Execute startup.sh command to start Tomcat

**/opt/apache-tomcat-9.0.76/bin/startup.sh**

* + Check whether Tomcat is started or not by executing below command

**ps -ef | grep tomcat**

* Add 8080 port in Security Group Inbound Rules
* Access **http://<PublicIP>:8080** from Browser
* Shutdown the tomcat

**/opt/apache-tomcat-9.0.76/bin/shutdown.sh**

# Jenkins Integration

In **tomcat server** Create the User, we use this user to copy the package from Jenkins Server to Tomcat Server

**adduser tomcatadmin**

**passwd tomcatadmin**

* + Provide the password twice (New Password and Retype New Password)
* Add the user in sudoers file to allow the user to become root user using sudo command

**echo "tomcatadmin ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers**

* Enable the Password authentication

**sed -ie 's/PasswordAuthentication no/PasswordAuthentication yes/' /etc/ssh/sshd\_config**

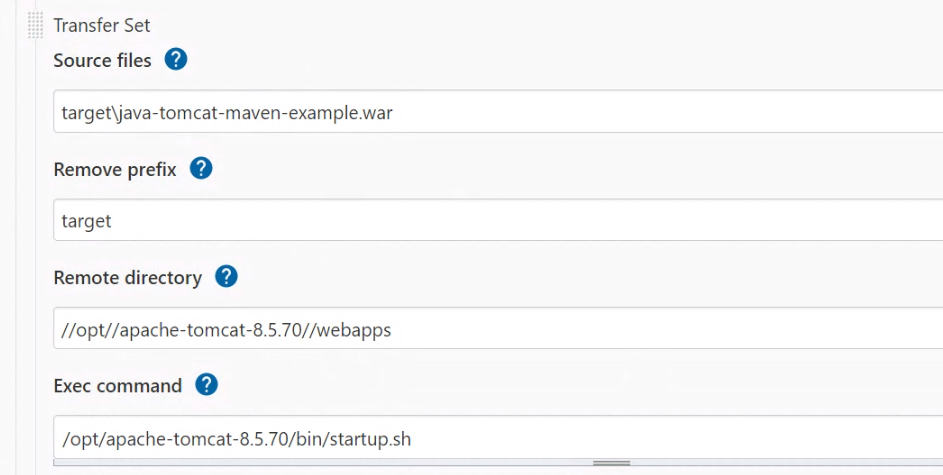
**service sshd restart**

* Let us have tomcatadmin user have permissions to copy into webapps folder and have permission to start the tomcat

**chmod -R 770 /opt/apache-tomcat-9.0.76**

**chgrp -R tomcatadmin /opt/apache-tomcat-9.0.76**

* Assuming that **Jenkins** is already installed and configured
  + If Jenkins is not installed already, please follow the instructions provided in Jenkins Installation document
* In **Jenkins Server**, Go to Jenkins Console
  + Mange Jenkins 🡪 Mange Plugins 🡪 Available
    - Install **Publish Over SSH** plugin
  + Go To Mange Jenkins 🡪 Configure System
    - At the bottom of the page, Click on Add to add ssh server
      * Name – Tomcat
      * Hostname – Private IP of Tomcat
      * Username – tomcatadmin
      * Click on Advanced
      * Check Use Password authentication
      * Provide the tomcatadmin user’s password
      * Click on Test Configuration to see if connection to Tomcat server is Success
* In **Jenkins Server**, Access the Console and Create the Freestyle Job
  + Integrate with Git
    - Git Repository URL
      * https://github.com/daticahealth/java-tomcat-maven-example.git
  + Integrate with Maven
  + Go to the Job 🡪 Configure 🡪 Add Post Build Step 🡪 Click on Send Build artifacts over SSH



In our case

**Source files -** target\java-tomcat-maven-example.war

**Remove Prefix -** target

**Remote Directory -** //opt//apache-tomcat-9.0.76//webapps

**Exec Command -** /opt/apache-tomcat-9.0.76/bin/startup.sh

* Save the Jon
* Build the Job
* Go To Tomcat Server
  + Go To Security group and make sure that you have added port no 8080 in Inbound Rules
  + Copy the Public IP address
  + Access it from browser
    - **http://<PublicIP of Tomcat Server>:8080/java-tomcat-maven-example**
    - You should see the page in your application running on a server

sudo yum install java-11-openjdk