



Katalon Studio For Automated Testing

Sesi 7



Jenkins on Google Cloud Platform

Materi ini hanya bersifat Optional untuk knowledge lebih jauh dan Environment : Cloud yang bisa teman-teman pelajari sendiri nantinya.

Sebagai Lanjutan untuk Jenkins pada Google Cloud Platform.

INTEGRATION JENKINS ON GCP

- 1A. In the Google Cloud Console, go to the **VM instances** page.
[Go to VM instances](#)
2. Select your project and click **Continue**.
3. Click **Create instance**.
4. Specify a **Name** for your instance. See [Resource naming convention](#).
5. Optional: Change the **Zone** for this instance. Compute Engine randomizes the list of zones within each region to encourage use across multiple zones. **Use zone jakarta**.
6. Select a **Machine configuration** for your instance.
7. In the **Boot disk** section, click **Change** to configure your boot disk. Unless you explicitly choose a different boot disk, if the name of the new instance matches the name of an existing persistent disk, then the existing persistent disk automatically attaches to the new instance as the boot disk. **Choose Ubuntu 18.04**
8. In the **Public images** tab, choose an operating system and version.
9. Click **Save** to confirm your boot disk options.
10. Select **Allow HTTP traffic** or **Allow HTTPS traffic** to permit HTTP or HTTPS traffic to the VM. When you select one of these, Compute Engine adds a network tag to your VM, which associates the firewall rule with the VM. Then, Compute Engine creates the corresponding ingress [firewall rule](#) that allows all incoming traffic on [tcp:80](#) (HTTP) or [tcp:443](#) (HTTPS).
11. Optional: If you chose an OS image that supports Shielded VM features, you can modify the [Shielded VM](#) settings:
 1. Click the **Security** tab in the **Management, security, disks, networking, sole tenancy** section.
 2. To enable [Secure Boot](#), select **Turn on Secure Boot**. Secure Boot is [disabled by default](#).
 3. To disable [vTPM](#), clear **Turn on vTPM**. vTPM is [enabled by default](#). Disabling vTPM also disables [integrity monitoring](#) because integrity monitoring relies on data gathered by [Measured Boot](#).
 4. To disable [integrity monitoring](#), clear the **Turn on Integrity Monitoring** checkbox. Integrity monitoring is [enabled by default](#).
12. Click the **Create** button to create and start the instance.

INTEGRATION JENKINS ON GCP

INSTALL & CONFIGURE JAVA SDK & JDK

Install JRE/JDK

You can install Open JDK which is packaged with Ubuntu 18.04 LTS by default.

```
sudo apt update
```

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

Once Java Runtime Environment is installed you can check the Java version with the command as follows.

```
java
```

You will see the output resembling this.

```
openjdk version "10.0.2" 2018-07-17
```

```
OpenJDK Runtime Environment (build 10.0.2+13-Ubuntu-1ubuntu018.04.4)
```

```
OpenJDK 64-Bit Server VM (build 10.0.2+13-Ubuntu-1ubuntu018.04.4, mixed mode)
```

Next you can install the default JDK to compile and run some specific software packages.

```
sudo apt install default-jdk
```

Once Java Development Kit is installed you can check the version with the `javac` compiler command as follows.

```
javac -version
```

You will get the version of JDK installed currently.

How to Install OpenJDK 8

Java 8 is also a Long Term supported version which has support until 2022 and widely supported. To install OpenJDK 8, execute the following command.

```
sudo apt install openjdk-8-jdk
```

Verify the installed version of Java.

```
java -version
```

Managing Java versions

You can configure which Java version to use as default.

```
sudo update-alternatives --config java
```

This is the output with all versions installed in your instance.

There are 3 choices for the alternative java (providing /usr/bin/java).

Selection	Path	Priority	Status

0	/usr/lib/jvm/java-11-openjdk-amd64/bin/java	1101	auto mode
1	/usr/lib/jvm/java-11-openjdk-amd64/bin/java	1101	manual mode
2	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081	manual mode

Press <enter> to keep the current choice[*], or type selection number:

Select the number associated with the Java version you need to use and hit **Enter**

Use the same `update-alternatives` command to get the installation path of your default Java version.

```
sudo update-alternatives --config java
```

Now we have 2 Java versions installed, their paths are

1. OpenJDK 11 is located at `/usr/lib/jvm/java-11-openjdk-amd64/bin/java`
2. OpenJDK 8 is located at `/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java`

Copy the installation path of your default version and add it in the `JAVA_HOME` environment variable.

```
sudo nano /etc/environment
```

At the end of this file, add the following line with your installation path. To use the official Java 8 by Oracle the variable will be as follows.

```
JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java"
```

Hit **Ctrl+X** followed by **Y** and **Enter** to save and exit the nano editor.

Now **JAVA_HOME** environment variable is set and available for all users.

Reload to apply changes.

```
source /etc/environment
```

To verify the environment variable of Java

```
echo $JAVA_HOME
```

You will get the installation path you just set.

INTEGRATION JENKINS ON GCP

OS : UBUNTU 18.04 LTS

Prerequisites

- A running Compute Engine,
- Installed Java 8

Setup Firewall Rules

Jenkins uses a custom port **8080** to run, so you need to create a firewall to all access to this port.

Go to VPC Network >> Firewall rules and click Create Firewall rules.

In **Name** enter `jenkins`

In **Targets** select `All instances in the network`

In **Source filter** select `IP ranges`

In **Source IP ranges** enter `0.0.0.0/0`

In **Protocols and ports** check **TCP** and enter `8080`

Click **Create**.

Install Jenkins

To install latest version of Jenkins, add the repository key to the system and add the repository address to the sources list.

```
sudo 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'
```

Now you can update and install Jenkins.

```
sudo apt update  
sudo apt install jenkins
```

Starting Jenkins

Once the installation is complete you can start Jenkins using the following command.

```
sudo systemctl start jenkins
```

You can also view the status of Jenkins using this command.

```
sudo systemctl status jenkins
```

If Jenkins is started successfully you will get a response similar to this.

Output

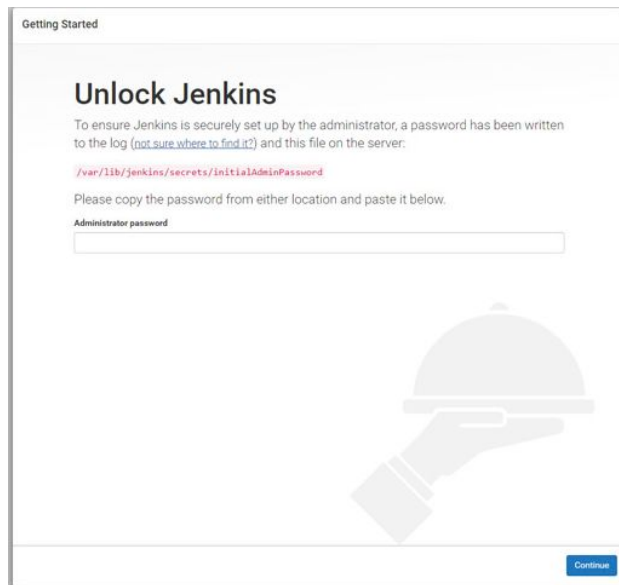
```
● jenkins.service - LSB: Start Jenkins at boot time  
   Loaded: loaded (/etc/init.d/jenkins; generated)  
   Active: active (exited) since Thu 2019-06-06 09:15:55 UTC; 46s ago  
     Docs: man:systemd-sysv-generator(8)  
    Tasks: 0 (limit: 1997)  
   CGroup: /system.slice/jenkins.service
```

Set Up Jenkins

Once everything is done you can open your browser and enter your IP address followed by the Jenkins port **8080**

The format will be like this **http://instance_external_ip:8080**

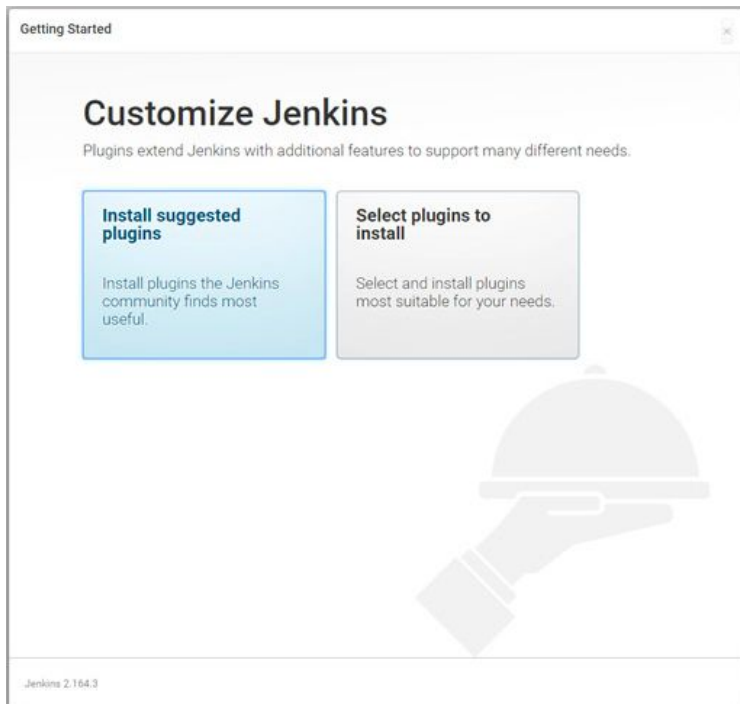
You will see the Unlock screen where you need to type the password to unlock Jenkins.



Execute the following command to get the password.

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

Copy the password and paste it in the **Administrator password** field to unlock and start the setup.



Click **Install suggested plugins** option to start the installation immediately.

Getting Started

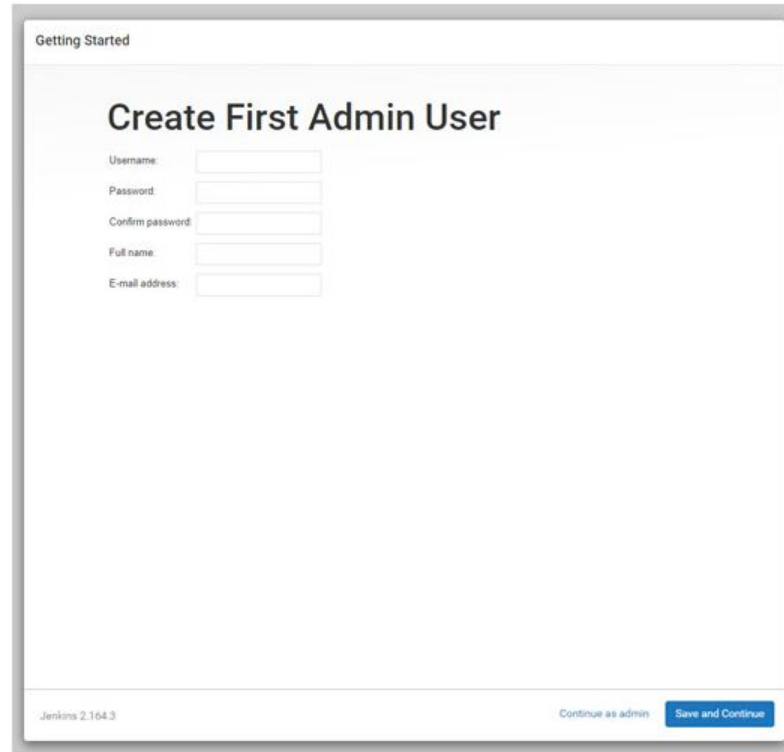
Getting Started

<input checked="" type="checkbox"/> Folders	<input checked="" type="checkbox"/> OWASP Markup Formatter	<input type="checkbox"/> Build Timeout	<input type="checkbox"/> Credentials Binding	Folders
<input type="checkbox"/> Timestampers	<input type="checkbox"/> Workspace Cleanup	<input type="checkbox"/> Ant	<input type="checkbox"/> Gradle	** JOK Tool
<input type="checkbox"/> Pipeline	<input type="checkbox"/> GitHub Branch Source	<input type="checkbox"/> Pipeline: GitHub Groovy Libraries	<input type="checkbox"/> Pipeline: Stage View	** Script Security
<input type="checkbox"/> Git	<input type="checkbox"/> Subversion	<input type="checkbox"/> SSH Slaves	<input type="checkbox"/> Matrix Authorization Strategy	** Command Agent Launcher
<input type="checkbox"/> PAM Authentication	<input type="checkbox"/> LDAP	<input type="checkbox"/> Email Extension	<input type="checkbox"/> Mailer	** Struts
				** Pipeline: Step API
				** SCM API
				** Pipeline: API
				** JUnit
				**ouncycastle API
				OWASP Markup Formatter

** - required dependency

Jenkins 2.164.3

Once the installation is complete you can create an admin user to login to the dashboard.



The screenshot shows the 'Getting Started' page of Jenkins. The main heading is 'Create First Admin User'. Below this, there are five input fields: 'Username', 'Password', 'Confirm password', 'Full name', and 'Email address'. At the bottom of the form, there is a 'Continue as admin' link and a 'Save and Continue' button. The footer of the page indicates 'Jenkins 2.164.3'.

Getting Started

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

Jenkins 2.164.3

[Continue as admin](#) [Save and Continue](#)

Finally you will see the Instance Configuration, you can use your domain name or IP address.

Click **Save and Finish**.