

Create a Master and Slave environment in Jenkins and run app deployment job in slave

Setting up a master-slave environment in Jenkins allows you to distribute build and deployment jobs across multiple machines.

Solution:

Connection Requirements:

Java must be same version installed in Master and Slave.

Installation:

1. EC2
2. Jenkins
3. Docker

EC2

- Create EC2 instance in AWS for installing Jenkins server.
- Login to AWS Management console



Sign in

☐ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☒ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

Account ID (12 digits) or account alias

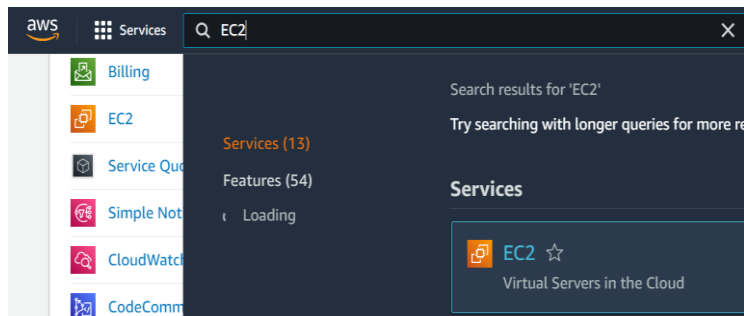
Next

By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

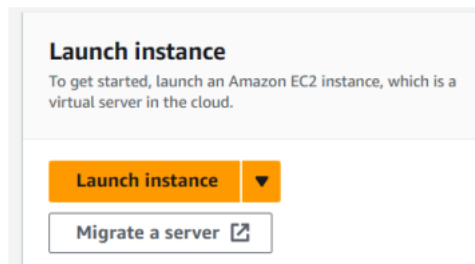
New to AWS?

Create a new AWS account

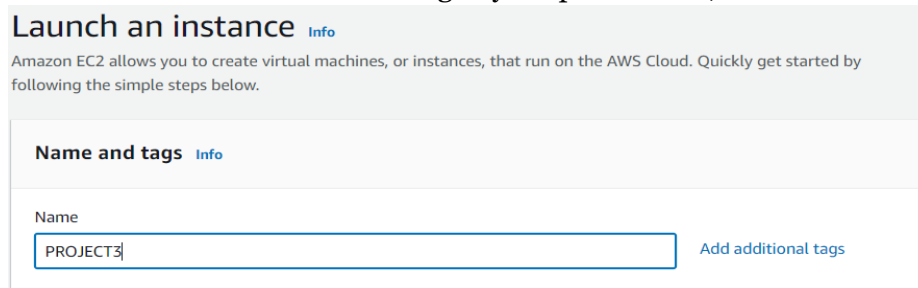
- Choose EC2 and click launch instance



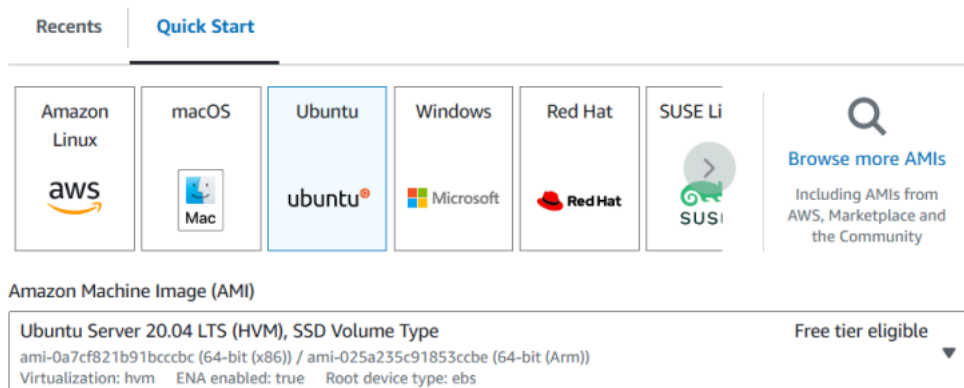
- Then under the EC2 management console we could able to find the Launch instance option, click that one for creating an instance:



- Then name the instance according to your preferences,



- Then we need to select an AMI Image for the instance, choose AMI image according to your preferences: Here I am selecting Ubuntu 20.04 AMI Image.



- Then we need to select the instance type

▼ **Instance type** [Info](#)

Instance type

t2.micro	Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true	
On-Demand Linux base pricing: 0.0124 USD per Hour	
On-Demand Windows base pricing: 0.017 USD per Hour	
On-Demand RHEL base pricing: 0.0724 USD per Hour	
On-Demand SUSE base pricing: 0.0124 USD per Hour	

[Additional costs apply for AMIs with pre-installed software](#)

- Then we need to select the keypair for security authentication purpose of the instance

▼ **Key pair (login)** [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Q Linux
×

[Create new key pair](#)

- Then under network settings, I am selecting default VPC, subnet no preference option, auto-assign public is enabled by default for default VPC & Subnet.

▼ **Network settings** [Info](#)

VPC - *required* [Info](#)

(default)
↻

Subnet [Info](#)

↻

[Create new subnet](#)

Auto-assign public IP [Info](#)

▼

- Configure storage and launch a instance.

▼ **Configure storage** [Info](#) Advanced

1x GiB Root volume (Not encrypted)

ⓘ Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
 ×

[Add new volume](#)

0 x File systems [Edit](#)

[EC2](#) > [Instances](#) > Launch an instance

✓ **Success**
 Successfully initiated launch of instance (i-09af4a63d3f83a318)

[▶ Launch log](#)

Jenkins

- Install a Jenkins in our EC2 machine.

Prerequisites

Minimum hardware requirements:

- 256 MB of RAM
- 1 GB of drive space (although 10 GB is a recommended minimum if running Jenkins as a Docker container)
- Jenkins requires Java to run, so we need to install java using this commands.

sudo apt update

```
ubuntu@ip-172-31-19-109:~$ sudo apt update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1149 kB]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [245 kB]
```

sudo apt install fontconfig openjdk-17-jre

```
ubuntu@ip-172-31-19-109:~$ sudo apt install fontconfig openjdk-17-jre
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
fontconfig is already the newest version (2.13.1-4.2ubuntu5).
fontconfig set to manually installed.
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf ca-certificates-java fonts-dejavu-extra java-common libasound2 libasound2-data libatk-wrappers-1.0
  openjdk-17-jre-headless
Suggested packages:
  default-jre libasound2-plugins alsa-utils pcscd libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
The following NEW packages will be installed:
  alsa-topology-conf alsa-ucm-conf ca-certificates-java fonts-dejavu-extra java-common libasound2 libasound2-data libatk-wrappers-1.0
  openjdk-17-jre openjdk-17-jre-headless
0 upgraded, 13 newly installed, 0 to remove and 21 not upgraded.
Need to get 51.2 MB of archives.
After this operation, 204 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 alsa-topology-conf all 1.2.5.1-2 [15.5 kB]
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libasound2-data all 1.2.6.1-1ubuntu1 [19.1 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libasound2 amd64 1.2.6.1-1ubuntu1 [390 kB]
Get:4 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 alsa-ucm-conf all 1.2.6.3-1ubuntu1.8 [43.3 kB]
Get:5 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 java-common all 0.72build2 [6782 B]
Get:6 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libpcsclite1 amd64 1.9.5-3ubuntu1 [19.8 kB]
Get:7 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jre-headless amd64 17.0.8.1+1-1ubuntu1 [43.3 kB]
Get:8 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 ca-certificates-java all 20190909ubuntu1.2 [11.1 kB]
Get:9 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 fonts-dejavu-extra all 2.37-2build1 [2041 kB]
Get:10 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libatk-wrapper-java all 0.38.0-5build1 [53.1 kB]
Get:11 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libgif7 amd64 5.1.9-2build2 [33.8 kB]
Get:12 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libgif7 amd64 5.1.9-2build2 [33.8 kB]
Get:13 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 openjdk-17-jre amd64 17.0.8.1+1-us1-0ubuntu1 [43.3 kB]
```

java -version

```
ubuntu@ip-172-31-19-109:~$ java -version
openjdk version "17.0.8.1" 2023-08-24
OpenJDK Runtime Environment (build 17.0.8.1+1-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 17.0.8.1+1-Ubuntu-0ubuntu122.04, mixed mode, sharing)
ubuntu@ip-172-31-19-109:~$
```

- Once java installation completes next start installing Jenkins using this commands.

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc
\https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
```

```
ubuntu@ip-172-31-19-109:~$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \
https://pkg.jenkins.io/debian binary/ | sudo tee \
/etc/apt/sources.list.d/jenkins.list > /dev/null
--2023-11-04 11:41:37-- https://pkg.jenkins.io/debian/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.78.133, 2a04:4e42:83::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|146.75.78.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.asc      100%[=====]
2023-11-04 11:41:37 (25.0 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]
```

sudo apt-get update sudo apt-get install Jenkins

```
sudo apt-get install jenkins
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Ign:5 https://pkg.jenkins.io/debian binary/ InRelease
Get:6 https://pkg.jenkins.io/debian binary/ Release [2044 B]
Get:7 https://pkg.jenkins.io/debian binary/ Release.gpg [833 B]
Get:8 https://pkg.jenkins.io/debian binary/ Packages [57.4 kB]
Fetched 60.3 kB in 1s (64.7 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 21 not upgraded.
Need to get 89.2 MB of archives.
After this operation, 90.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-lubuntu5 [204 kB]
Get:2 https://pkg.jenkins.io/debian binary/ jenkins 2.430 [89.0 MB]
Fetched 89.2 MB in 5s (16.5 MB/s)
Selecting previously unselected package net-tools.
(Reading database ... 124560 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-lubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-lubuntu5) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../archives/jenkins_2.430_all.deb ...
Unpacking jenkins (2.430) ...
Setting up net-tools (1.60+git20181103.0eebece-lubuntu5) ...
Setting up jenkins (2.430) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service -> /lib/systemd/system/jenkins.service.
Processing triggers for man-db (2.10.2-1) ...
```

- Once Jenkins installation finished we need to open port (8080) for accessing our Jenkins using aws security groups.
- Go to instance security groups and edit inbound rules and add 8080 with custom ip range and save.

Edit inbound rules [info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Security group rule ID	Type info	Protocol info	Port range info	Source info	Description - optional info	
sgr-08f00871e2c3358	All traffic	All	All	Custom	0.0.0.0/0	Delete
sgr-0da8db1700dfa7ab	Custom TCP	TCP	8080	Custom	0.0.0.0/0	Delete

- Next Accessing our Jenkins server using our instance public ip follows with port 8080.

<http://IP:8080>



- Getting started Jenkins page will open paste our admin password.
- Use this command to get Jenkins admin password

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

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

Continue

```
ubuntu@ip-172-31-19-109:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
b8d4fff8d5ce460089f8e6f191ac8115
ubuntu@ip-172-31-19-109:~$
```

- Copy and paste the password from terminal Jenkins page next click install suggested plugins.

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.430

- It will install all plugins

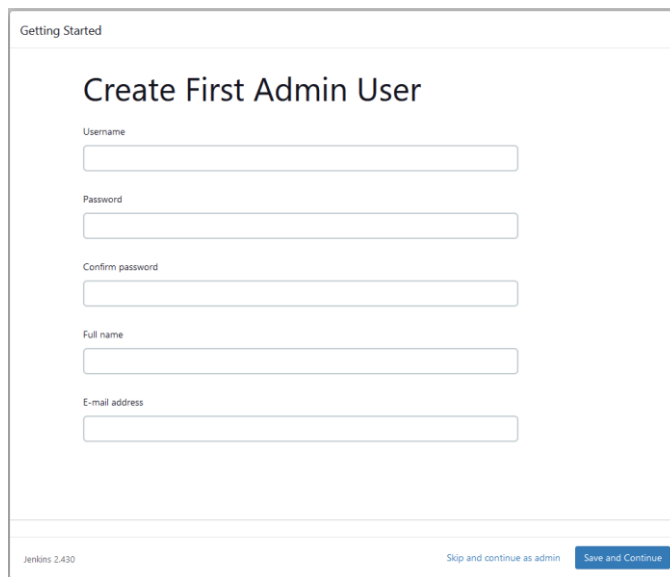
Getting Started

Getting Started

<input checked="" type="checkbox"/> Folders	<input checked="" type="checkbox"/> OWASP Markup Formatter	<input checked="" type="checkbox"/> Build Timeout	<input checked="" type="checkbox"/> Credentials Binding	<input checked="" type="checkbox"/> Icons API
<input checked="" type="checkbox"/> Timestampers	<input type="checkbox"/> Workspace Cleanup	<input type="checkbox"/> Ant	<input type="checkbox"/> Gradle	<input type="checkbox"/> Folders
<input type="checkbox"/> Pipeline	<input type="checkbox"/> GitHub Branch Source	<input type="checkbox"/> Pipeline: GitHub Groovy Libraries	<input type="checkbox"/> Pipeline: Stage View	<input type="checkbox"/> OWASP Markup Formatter
<input type="checkbox"/> Git	<input type="checkbox"/> SSH Build Agents	<input type="checkbox"/> Matrix Authorization Strategy	<input type="checkbox"/> PAM Authentication	<input type="checkbox"/> Struts
<input type="checkbox"/> LDAP	<input type="checkbox"/> Email Extension	<input type="checkbox"/> Mailer		<input type="checkbox"/> Bouncycastle API
				<input type="checkbox"/> Instance Identity
				<input type="checkbox"/> JavaBeans Activation Framework (JAF) API
				<input type="checkbox"/> JavaMail API
				<input type="checkbox"/> Pipeline: Step API
				<input type="checkbox"/> Token Macro
				<input type="checkbox"/> Build Timeout
				<input type="checkbox"/> Credentials
				<input type="checkbox"/> Plain Credentials
				<input type="checkbox"/> Trilead API
				<input type="checkbox"/> SSH Credentials
				<input type="checkbox"/> Credentials Binding
				<input type="checkbox"/> SCM API
				<input type="checkbox"/> Pipeline: API
				<input type="checkbox"/> commons-lang3 v3.x Jenkins API
				<input type="checkbox"/> Timestampers
				<input type="checkbox"/> Caffeine API
				<input type="checkbox"/> - required dependency

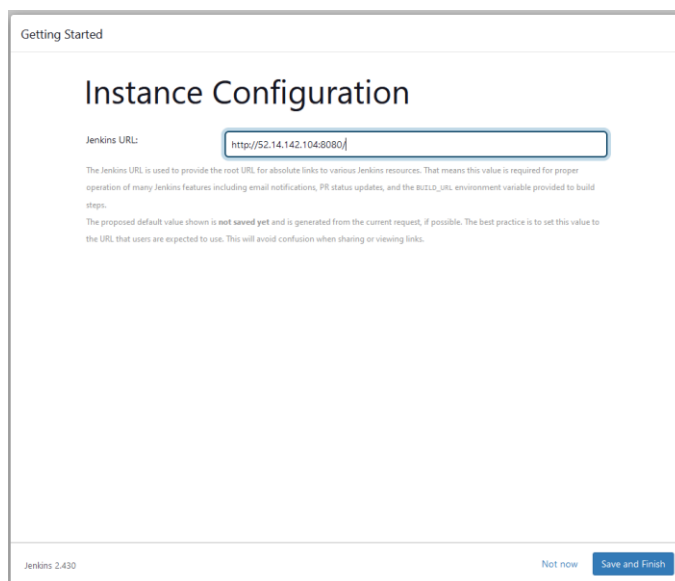
Jenkins 2.430

- Create a user to work with Jenkins or continue as admin.



The screenshot shows the 'Getting Started' page for Jenkins 2.430. The main heading is 'Create First Admin User'. Below the heading are five input fields: 'Username', 'Password', 'Confirm password', 'Full name', and 'E-mail address'. At the bottom right, there are two buttons: 'Skip and continue as admin' and 'Save and Continue'.

- Click save and continue to enter Jenkins dashboard.



The screenshot shows the 'Getting Started' page for Jenkins 2.430. The main heading is 'Instance Configuration'. Below the heading is a 'Jenkins URL' input field with the value 'http://52.14.142.104:8080/'. Below the input field is a paragraph of text explaining the Jenkins URL and its purpose. At the bottom right, there are two buttons: 'Not now' and 'Save and Finish'.

- Go to Manage Jenkins >> Plugins >> install docker and pipeline plugins.
- Next use the following commands to install docker in master as well in slave machine also.

sudo apt update
sudo apt install docker.io

```
ubuntu@ip-172-31-10-49:~$ sudo apt update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
```

```
ubuntu@ip-172-31-10-49:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd dns-root-data dnsmasq-base docker.io pigz runc ubuntu-fan
0 upgraded, 8 newly installed, 0 to remove and 54 not upgraded.
Need to get 69.7 MB of archives.
After this operation, 267 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

docker info

```
ubuntu@ip-172-31-10-49:~$ docker info
Client:
 Version:      24.0.5
 Context:      default
 Debug Mode:   false
```

- Next Create a slave machine in Jenkins using manage Jenkins and go to nodes
- Name the slave and click create.

Dashboard > Manage Jenkins > Nodes > New node

New node

Node name

Type

☒ Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

[Create](#)

Dashboard > Manage Jenkins > Nodes >

Name [?](#)

Description [?](#)

[Plain text](#) [Preview](#)

Number of executors [?](#)

Remote root directory [?](#)

- Give the Label name for our slave that label will be used in all our codes.
- Choose Launch agent by controlling it to the container.
- Choose WebSocket and select availability keep agent as possible.

Labels ?

agent

Usage ?

Use this node as much as possible

Launch method ?

Launch agent by connecting it to the controller

☐ Disable WorkDir ?

Custom WorkDir path ?

Internal data directory ?

remoting

☐ Fail if workspace is missing ?

☒ Use WebSocket ?

☒ Use WebSocket ?

Advanced ?

Availability ?

Keep this agent online as much as possible

Node Properties

☐ Disable deferred wipeout on this node ?

☐ Environment variables

☐ Tool Locations

Save

- Check the configurations and click save it will create some file and command for our Jenkins slave connection.
- Every time instance stopped the ip will be changed so you need to change the IP address of your Jenkins.
- Go to Manage Jenkins >> system >> Change the Jenkins URL with a new IP address.

Dashboard > Manage Jenkins > Nodes > Slave

Agent Slave Mark this node temporarily offline ?

Delete Agent Configure Build History Load Statistics Log

Run from agent command line: (Unix) ?

```
curl -sO http://18.188.178.214:8080/jnlpJars/agent.jar
java -jar agent.jar -jnlpUrl http://18.188.178.214:8080/computer/Slave/jenkins-agent.jnlp -workDir "/home/ubuntu/build"
```

Add description

- Go to the /home/Ubuntu directory and paste the commands provided by Jenkins master.
- The code will be download a jar file it will used to create a connection with our Jenkins master.

```
aws Services Search [Alt+S]
ubuntu@ip-172-31-0-116:~$ curl -sO http://18.188.178.214:8080/jnlpJars/agent.jar
ubuntu@ip-172-31-0-116:~$ java -jar agent.jar -jnlpUrl http://18.188.178.214:8080/computer/Slave/jenkins-agent.jnlp -workDir "/home/ubuntu/build"

ubuntu@ip-172-31-0-116:~$ curl -sO http://18.188.178.214:8080/jnlpJars/agent.jar
ubuntu@ip-172-31-0-116:~$ java -jar agent.jar -jnlpUrl http://18.188.178.214:8080/computer/Slave/jenkins-agent.jnlp -workDir "/home/ubuntu/build"
Dec 14, 2023 6:30:53 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/build/remoting as a remoting work directory
Dec 14, 2023 6:30:53 AM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to /home/ubuntu/build/remoting
Dec 14, 2023 6:30:54 AM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: Slave
Dec 14, 2023 6:30:54 AM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 3160.vd76b_9ddd10cc
Dec 14, 2023 6:30:54 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using /home/ubuntu/build/remoting as a remoting work directory
Dec 14, 2023 6:30:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: WebSocket connection open
Dec 14, 2023 6:30:55 AM hudson.remoting.jnlp.Main$CuiListener status
INFO: Connected
```

Dashboard > Manage Jenkins > Nodes > Slave

Status
 Delete Agent
 Configure
 Build History
 Load Statistics
 Script Console
 Log
 System Information
 Disconnect

Agent Slave

Agent is connected.

Labels

agent

Projects tied to Slave

None

- Check logs of our Slave node it shows connected and online state.

Dashboard > Nodes > Slave > Log

Status
 Delete Agent
 Configure
 Build History
 Load Statistics
 Script Console
 Log
 System Information
 Disconnect

```

Inbound agent connected from 18.223.172.35
Remoting version: 3160.vd76b_9ddd10cc
Launcher: JNLPLauncher
Communication Protocol: WebSocket
This is a Unix agent
Agent successfully connected and online

```

- Check nodes it will be attached to our master.

Nodes + New Node Node Monitoring 🔄

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
	Built-in Node	Linux (amd64)	In sync	2.74 GB	0 B	2.74 GB	0ms
	Slave	Linux (amd64)	In sync	5.35 GB	0 B	5.35 GB	133ms
Data obtained		1.3 sec	1.4 sec	1.3 sec	1.3 sec	1.3 sec	1.3 sec

- Create a Project as pipeline project and configure it.

Enter an item name

jenkinsnode

» Required field

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

OK

Branch Pipeline

- Give our repo URL for declarative checkout..
- Choose Pipeline script from SCM and give our repo URL.

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

×

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

Script Path ?

Jenkinsfile

☒ Lightweight checkout ?

[Pipeline Syntax](#)

Save

Apply

- Change our repository branch and Jenkins file for reference.
- Click save and apply and click build now button to start our job.

Dashboard > new >

Status

new

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

GitHub

Rename

Pipeline Syntax

Build History

trend ▾

Filter builds...

Dec 14, 2023, 8:07 AM

Stage View

	Declarative: Checkout SCM	Build	Deploy
Average stage times: (Average full run time: ~4s)	321ms	588ms	2s
Dec 14 13:37 1 complete	321ms	588ms	2s

Permalinks

- Last build (#9), 15 sec ago
- Last stable build (#9), 15 sec ago

Add description

Disable Project

