1. Install and configure Jenkins server

### Configure A Docker Host:

1. Setup Docker host in another machine and install all prerequire docker packages.
2. Enable API in your docker host.
3. Open the below file and do the modification.

sudo vi /lib/systemd/system/docker.service

1. modify:

ExecStart=/usr/bin/docker daemon -H fd:// -H tcp://0.0.0.0:4243

1. Make sure docker service notices the modification.

systemctl daemon-reload

1. Restart service:

sudo service docker restart

1. Test your API from another machine terminal.

curl <http://localhost:4243/version>

{"Version":"1.12.3","ApiVersion":"1.24","GitCommit":"6b644ec","GoVersion":"go1.6.3","Os":"linux","Arch":"amd64","KernelVersion":"4.4.0-42-generic","BuildTime":"2016-10-26T22:01:48.986273588+00:00"}

Create A Jenkins Slave Docker Image:

Next step is to create a slave image. The image should contain the following minimum configurations to act as a slave.

1. sshd service running on port 22.

2. Jenkins user with password.

3. All the required application dependencies for the build. For example, for a java maven project, you need to have git, java, and maven installed on the image.

I have created a [Jenkins image for maven](https://hub.docker.com/r/bibinwilson/jenkins-slave/). You can use this image or use its [Dockerfile](https://github.com/bibinwilson/jenkins-docker-slave) a reference for creating your own.

Make sure sshd service is running and can be logged into the containers using a username and password. Otherwise, Jenkins will not be able to start the build process.

### Configure Jenkins Server

1. Head over to Jenkins Dashboard –> Manage jenkins –> Manage Plugins.

2. Under available tab, search for “Docker Plugin” and install it.

3. Once installed, head over to jenkins Dashboard –> Manage jenkins –>Configure system.

4. Under “Configure System”, if you scroll down, there will be a section named “cloud” at the last. There you can fill out the docker host parameters for spinning up the slaves.

5. Under docker, you need to fill out the details as shown in the image below.

Note: Replace “Docker URL” with your docker host IP. You can use the “Test connection” to test if jenkins is able to connect to the docker host.



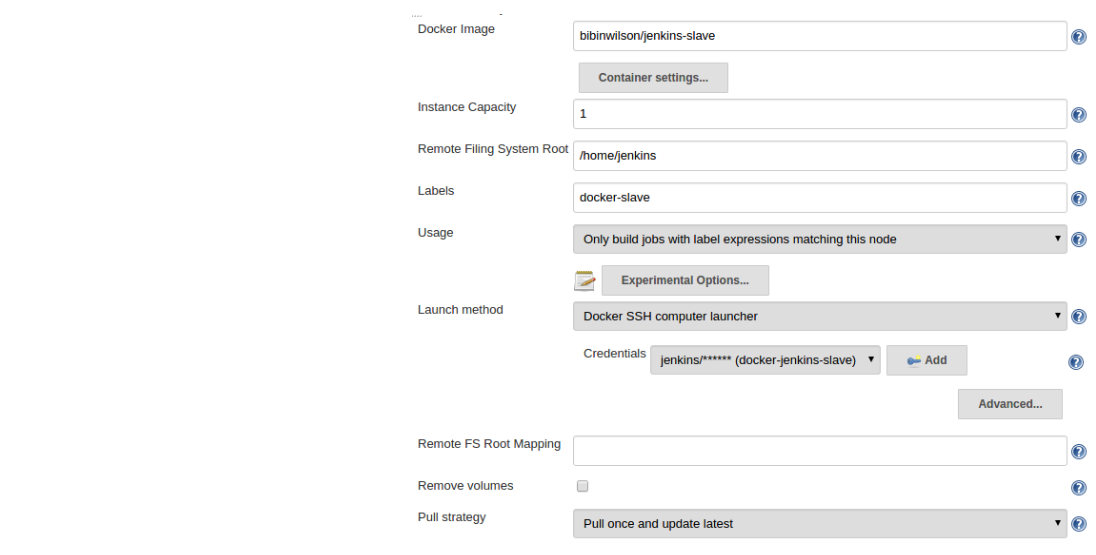
5. Now, from “**Add Docker Template**” dropdown, click “**docker** **template**” and fill in the details based on the explanation and the image given below.

**Docker Image** – Image that you created for the slave.

**Remote Filing System Root** – Home folder for the user you have created. In our case it’s jenkins.

**Labels** – Identification for the docker host. It will be used in the Job configuration.

**Credentials** – click add and enter the username and password that you have created for the docker image. Leave the rest of the configuration as shown in the image below and click save.



### Building Jobs On Docker Slaves