

DevOps

Brief Introduction

This is a Step-by-step tutorial.

After this tutorial, you will be able to:

1. Create a docker server on your server.
2. Deploy NodeJs application to Docker container via Jenkins.
3. Auto deploy application when you push code to your git repo.

Prerequisites

1. A server: Local pc (for practice) or VPS
2. Basic knowledge of Docker & Jenkins
3. Linux command (basic)

Install

Install Docker

You could skip this step if you have already have the docker server installed in your server. Or you could uninstall the old one, then follow this guide.

You could check the installation guide @ [Docker Document](#)

As most of you use CentOS, I will introduce the step for install docker on CentOS, for other system, please Kindly check above Docker document link.

There is two ways to install Docker on CentOS,

1. Install from repository
2. Install from a package

Here we use the first way to do it.

Firstly, we need to set up the repository:

1. Install required packages:

```
sudo yum install -y yum-utils \ device-mapper-persistent-data \ lvm2
```

2. Set up the stable repository:

```
sudo yum-config-manager \ --add-repo \ https://download.docker.com/linux/centos/docker-ce.repo
```

Then, install Docker CE

```
sudo yum install docker-ce
```

Finally, Startup Docker

```
sudo systemctl start docker
```

Install & Run Jenkins

Firstly, let's install the jenkins docker image

```
sudo docker pull jenkins
```

Then, we need to create a jenkins work directory

```
mkdir /var/jenkins_node
```

We also need to grant the operation access to the jenkins user

```
sudo chown -R 1000 /var/jenkins_node
```

Finally, you should start up a jenkins container via jenkins images

```
docker run -d --privileged=true --name myjenkins -p 49001:8080 -v /var/jenkins_node:/var/jenkins_home jenkins
```

After this, you could use `docker ps -a` to check whether your jenkins container is up or not, If not, you could use `docker logs <containerID>` to check the error log

As a result, when you visit `http://<your server ip>:49001`, you will get below screen

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/jenkins_home/secrets/initialAdminPassword
```

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

Administrator password



Continue

Then, let's find the initial password and fill it it.

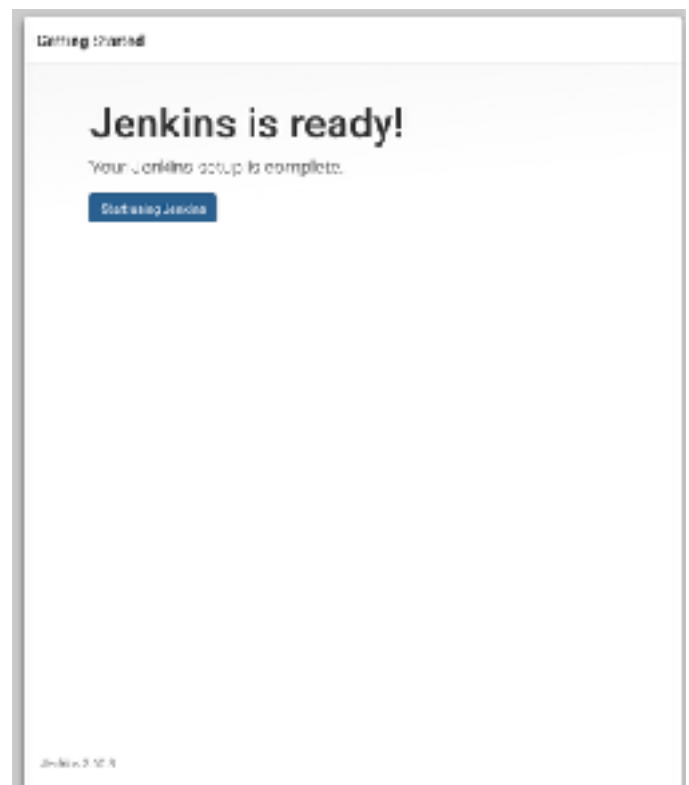
```
cat /var/jenkins_node/secrets/initialAdminPassword
```

And we could install the suggested plugins



You should create a first admin account, then the installation of Jenkins is done.

This screenshot shows the 'Create First Admin User' screen in the Jenkins installation wizard. The title is 'Create First Admin User'. There are five input fields: 'USERNAME' (highlighted with a blue border), 'PASSWORD', 'PASSWORD2', 'FIRST NAME', and 'LAST NAME'. At the bottom, there is a 'Continue as admin' link and a 'Show web console' button. The bottom left corner shows 'Jenkins 2.68.0'.



Connect your VPS server via jenkins over ssh
 Firstly we should install the 'Publish over SSH' plugin in jenkins

名称	版本
Publish Over SSH Send build artifacts over SSH	1.19.1
SSH Execute shell scripts on remote host using ssh (pre and post build). Based on the cool scp plugin.	2.6.1
Distributed Fork Turns a Jenkins cluster into a general purpose batch job execution environment through an SSH-like CLI. This plugin adds a new command 'distfork' to Jenkins CLI, which can be used to execute arbitrary command on a agent of your choice. The distfork command is modeled after ssh, but it's Jenkins aware — for example, instead of hardcoding a machine name, you can specify a label to let Jenkins choose a agent. This opens up a Jenkins cluster for doing all kinds of things without requiring a job/build action.	1.7
SSH Agent	1.15
SSH2 Easy This plugin allows you to ssh2 remote server to execute linux commands , shell , sftp upload, download etc	1.4
Terminate ssh processes	1.0

Once the plugin is done for the installation, the jenkins will restart.
 Then let's try to connect to your VPS via the plugin

If you got success after click 'test configuration', it means it could reach your VPS.
Then, click 'save'

The screenshot shows the 'Publish over SSH' configuration page in Jenkins. The 'Publish over SSH' tab is selected and highlighted with a red box. The 'Jenkins SSH Key' section contains fields for 'Passphrase', 'Path to key', and 'Key'. The 'SSH Servers' section contains a table with one entry: 'VPS-tokyo'. The 'Name' field is 'VPS-tokyo', 'Hostname' is '46.82.42.0', 'Username' is 'root', and 'Remote Directory' is '/'. The 'Test Configuration' button is highlighted with a red box, and the 'Success' message is also highlighted with a red box. The 'Save' button is highlighted with a red box.

1. enter your VPS login password

2. enter IP of your VPS

3. enter the password of it

保存

Install basic docker image for NodeJs

Since it's a NodeJs project, so it's necessary to install a basic NodeJs image via jenkins.
This is the test git repo: <https://github.com/MeatPieYan/BasicNodeJsDockerImage>

The screenshot shows the 'New Item' page in Jenkins. The 'New Item' button is highlighted with a red box. The 'Enter an item name' field is highlighted with a red box and contains the text 'NodeImageBasic'. The 'Build a free-form software project' option is highlighted with a red box.

新建

Enter an item name

NodeImageBasic

构建一个自由风格的软件项目

General 运行配置 构建策略 构建环境 构建 构建后操作

项目名称:

描述:

[\[Plugin icon\]](#) [\[Link\]](#)

☐ GitHub project
☐ The code is in
☐ 是否忽略构建
☐ 参数化构建过程
☐ 延迟构建
☐ 若必要则部署并安装

[高级...](#)

源码管理

☐ Maven
☒ **Git** 1. choose 'Git', and enter the repo url & credentials

Repositories

Repository URL:

Credentials: [Add](#)

[Add Repository](#)

Branches to build

pattern (optional) (regex for 'any') [Add Branch](#)

源仓库目录:

Additional References: [Add](#)

☐ Subversion

构建触发器

☐ 触发器配置 (详细帮助)
☐ Build after other projects are built
☐ Build periodically
☐ GitHub hook trigger for GITPoll polling
☐ Poll SCM

构建环境

☐ Delete workspace before build starts
☐ Use container (e.g. Docker)
☐ Send build console commands over SSH before the build starts
☒ Send files or execute commands over SSH after the build runs

SSH Publishers

SSH Name:

Transfer

Transfer Dir:

Source File:

Remote profile:

Remote directory:

Exec command:

[Add Transfer Set](#)

[Add SSH Publisher](#)

☐ Abort the build if it's stuck
☐ Add timestamps to the console output
☐ With Art

构建

构建策略: [\[Link\]](#)

构建后操作

构建后操作: [\[Link\]](#)

[\[Link\]](#) [\[Link\]](#)

<== enter Git related info

<== the script is as below

```
cd /var/jenkins_node/
workspace/NodeImageBasic
\
&& docker build --rm --no-
cache=true -t
node_pm2:8.10-2.10 - <
Dockerfile
```

After save it, you could try to run it.



Once the build is successful, we could check whether we have create a basic image in docker
If build status is unstable or failure, you could check the console of this build.

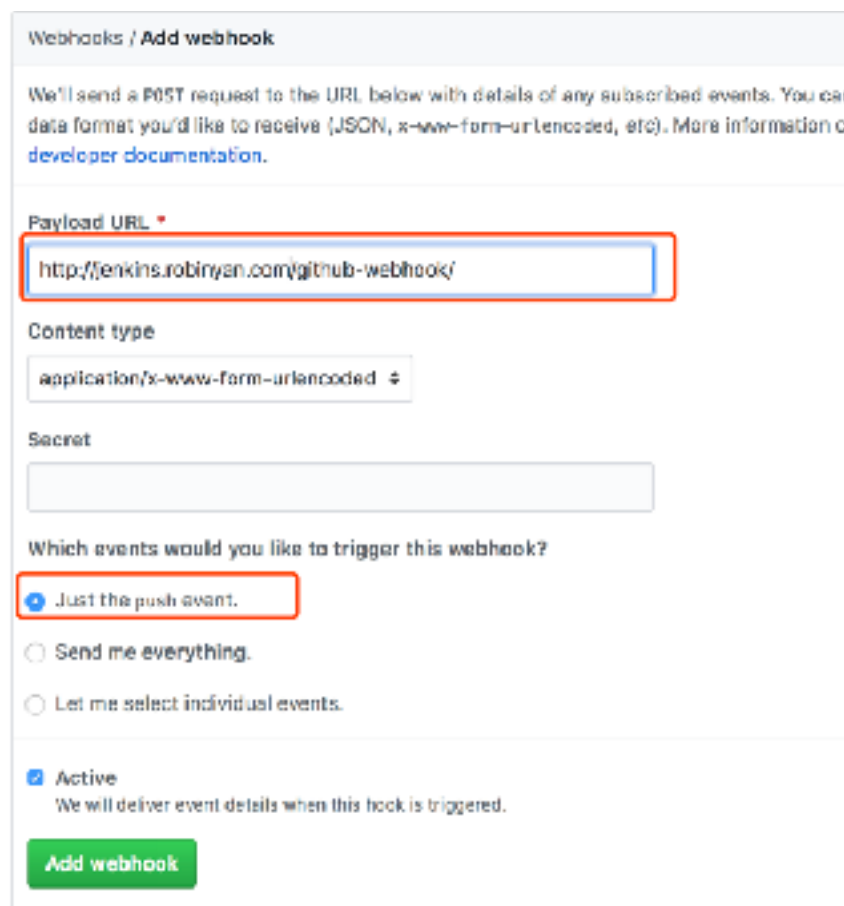
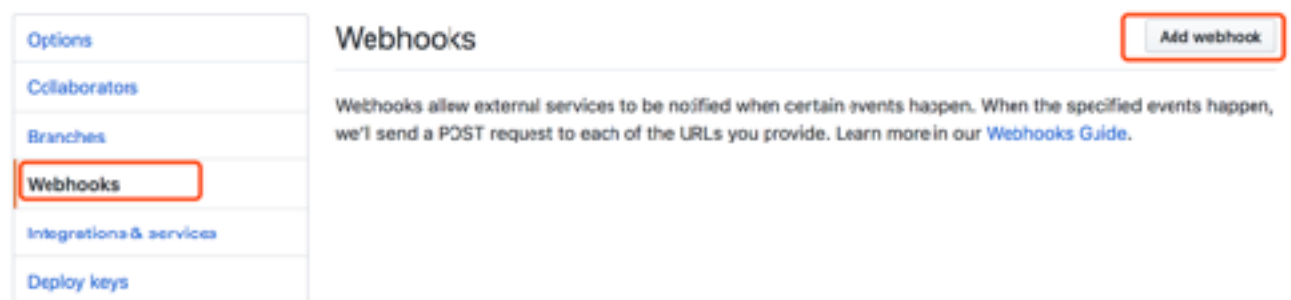
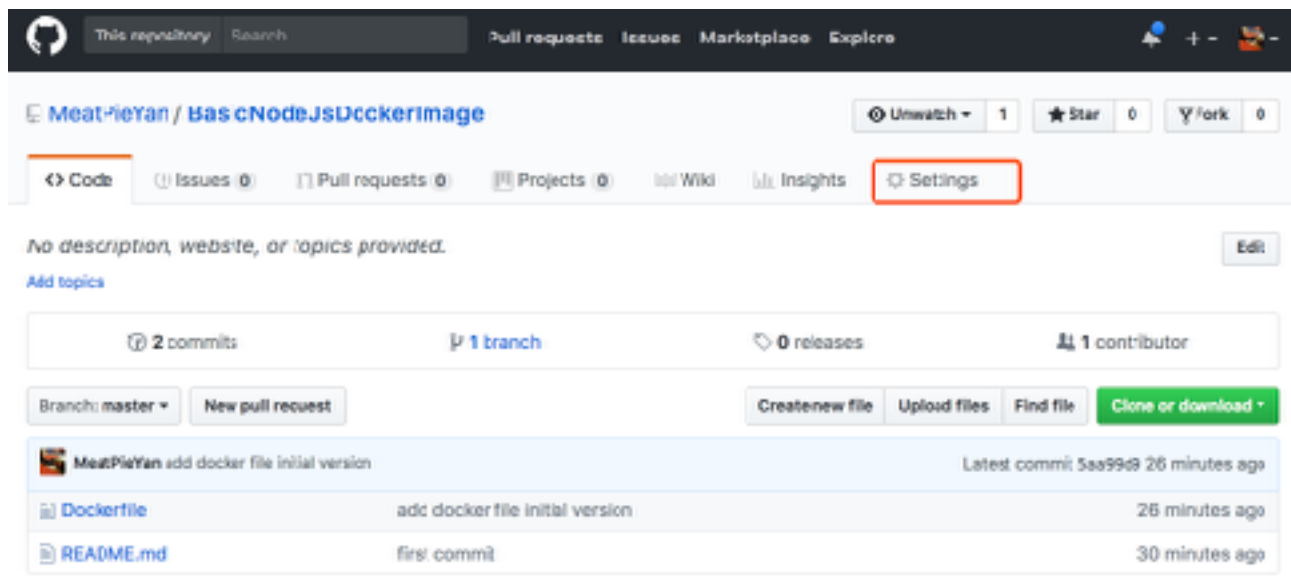


```
root@vultr:~# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
node pm2	8.10-2.10	6b2cb/c60/8a	9 minutes ago	94.3MB
jenkins	latest	7b210b6c238a	3 weeks ago	801MB
node	8.10-alpine	adc4b0f5bc53	4 weeks ago	68.1MB

Install basic docker image for NodeJs automatically

As of now, we have done image creation, but if we want to update our basic image, how shall we do it. We could update the dockerfile in the git repo, and trigger the jenkins task again. But can we do it automatically? The answer is yes.

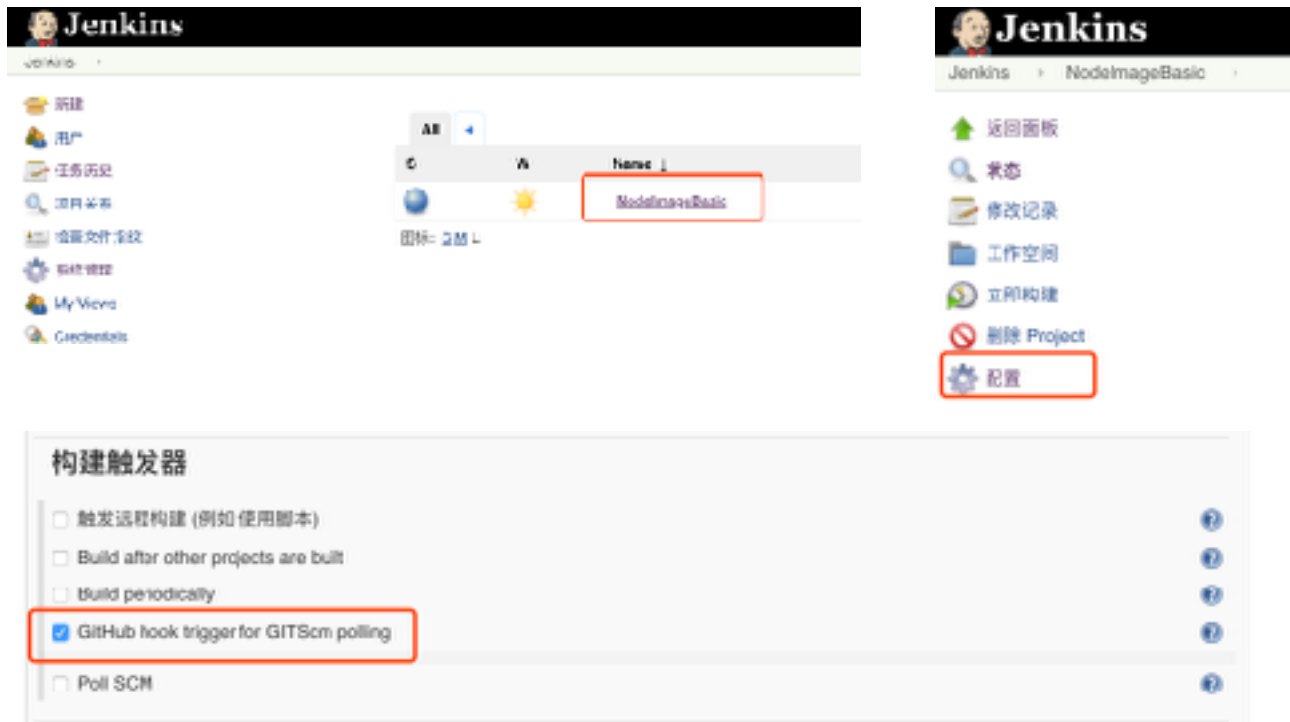


<== enter the url like this:

http://<you vps id>:<jenkins port>/github-webhook

If you have a domain like me, you could also enter the domain instead of ip:port

For Jenkins, we need to do some change to make it effective.



let's have a try with it
once you push any update into your git repo, a new Jenkins build will be triggered.



Seems everything is ok, then let's build our project.

Project deployment

Brief introduction

After this section, we need to deploy the project into docker container

Create Jenkins task

Enter an item name

Node_Test

* Required field

-  **构建一个自由风格的软件项目**
这是Jenkins的主要功能,Jenkins将会结合任何SCM和任何构建系统来构建你的项目,甚至可以...
-  **Pipeline**
Orchestrates long-running activities that can span multiple build slaves. Suitable for building p... and/or organizing complex activities that do not easily fit in free-style job type.
-  **构建一个多配置项目**
适用于多配置项目,例如多环境测试,平台指定构建,等等.
-  **文件夹**
创建一个可以嵌套存储的容器。利用它可以进行分组。视图仅仅是一个过滤器,而文件夹则是... 相同名称的内容。只要它们在不同的文件 夹里即可。
-  **GitHub Organization**
Scans a GitHub organization (or user account) for all repositories matching some defined ma...
-  **Multibranch Pipeline**
Creates a set of Pipeline projects according to detected branches in one SCM repository.

if you want to create a new item from other existing, you can use this option:

 Copy from

OK

Everything of this task is same like the one to build basic docker image except the script, below is the script. For git web hook, also the same.

```
docker stop container || true \
  && docker rm container || true \
  && cd /var/jenkins_node/workspace/Node_Test \
  && ls \
  && docker build --rm --no-cache=true -t node_project_img . \
  && docker run -d --name container -p 3000:3000 -v /var/jenkins_node/workspace/Node_Test:/
home/project node_project_img
```

项目名称

描述

Project_Team

☐ GraphQL product
 ☐ Throttle backls
 ☐ 系统启动构建
 ☐ 非结构化构建过程
 ☐ API构建
 ☐ 在部署时实时更新构建

2002年12月

☐ None

☒ Git

Repositories

Repository URL:

Credentials:

Branches to build

Branch Specifier (Pattern for "git")

Build configuration

Additional Environment

☐ Subversion

构建触发器

- ☐ Build after other projects are built
- ☐ Build periodically
- ☒ Build when triggered by Git commit
- ☐ Build SCM

构建环境

☐ Delete workspace before build (Auto)
☐ Use secret(s) or file(s)
☐ Shell task or execute commands (over SSH) before the build starts
☐ Shell task or execute commands (over SSH) after the build runs
☒ Serial task or execute commands (over SSH) after the build runs

SSH Server

Name

Transfers

Transfer Set

Source files

Remote paths

Remote directory

Exec command

```
docker stop container if true \
  && docker rm container if true \
  && cd /usr/jenkins/code/workspaceMode_Test \
  && ls \
  && docker build -f /Dockerfile \
  && docker tag -s
```

Other **Script**, **Exec** command or both must be supplied

All of the transfer fields (except the From Internet) support substitution of `$(VARNAME)` or `$(VARNAME:regex)`

Add Transfer Set

Add Server

构建

增加此項月份

构建后操作

植物构造是操作变量。