

Taiga

Other ways to setup initial environment

Table of Contents

- 1. Setup using **taiga-scripts** 1
 - 1.1. Introduction 1
 - 1.2. Important notes about **taiga-scripts** 1
 - 1.3. Requirements 1
 - 1.4. Setup server-like environment 1
 - 1.5. Setup development-like environment 1
 - 1.6. Upgrade from Ubuntu 16.04 taiga installation to Ubuntu 16.04 2
- 2. Setup using **taiga-vagrant** 3
 - 2.1. Introduction 3
 - 2.2. Important notes about using **taiga-vagrant** 3
 - 2.3. Dependencies 3
 - 2.4. Installation 3
- 3. Setup using **taiga-docker** 3
 - 3.1. Important notes about using **taiga-docker** 4

1. Setup using taiga-scripts

1.1. Introduction

taiga-scripts primarily follows the standard installation process, but makes it more automatic. If something goes wrong in the process, you should try to fix it by following the manual deploy process.

taiga-scripts always installs a specific (the most recent) version of Taiga (taiga-front and taiga-back)

1.2. Important notes about taiga-scripts

- In an alpha state, and has not been heavily tested in different environments.
- Designed to work with taiga-vagrant (see next section of this documentation).
- Does not support reprovisioning (updating) at this moment, that is still in development.
- Does not support installation with root user.
- The installation is not 100% ready for production.

1.3. Requirements

- Freshly installed and updated Ubuntu 16.04
- Bash shell
- Git

1.4. Setup server-like environment

Make sure you have a fresh and updated Ubuntu 16.04 machine or virtual image:

```
$ git clone https://github.com/taigaio/taiga-scripts.git
$ cd taiga-scripts
$ bash setup-server.sh
```

1.5. Setup development-like environment

This setup is much like the previous one but with a few differences:

- It does not install services like nginx, as they are only necessary for production
- It sets all hostnames to localhost to easily run everything on localhost

Make sure you have a fresh and updated Ubuntu 16.04 machine or virtual image:

```
$ git clone https://github.com/taigaio/taiga-scripts.git
$ cd taiga-scripts
$ bash setup-devel.sh
```

1.6. Upgrade from Ubuntu 16.04 taiga installation to Ubuntu 16.04

To upgrade from 14.04 to 16.04 you have to follow the official Ubuntu process:

```
$ sudo apt-get install update-manager-core
$ sudo do-release-upgrade -p
```

Then you have to upgrade to the new postgres version maintaining the old data:

```
$ sudo apt-get install -y postgresql-9.5 postgresql-contrib-9.5
$ sudo apt-get install -y postgresql-doc-9.5 postgresql-server-dev-9.5
$ sudo pg_dropcluster 9.5 main --stop
$ sudo pg_upgradecluster 9.3 main
$ sudo reboot
```

Then you have to use the circus service from ubuntu.

```
$ pip uninstall circus pyzmq
$ apt-get install circus
$ cp ~/conf/circus.ini /etc/circus/circusd.conf
```

Rebuild your virtualenv

```
$ rmvirtualenv taiga
$ mkvirtualenv taiga -p /usr/bin/python3.5
$ cd ~/taiga-back
$ pip install -r requirements.txt
```

Reboot the system

```
$ sudo reboot
```

2. Setup using taiga-vagrant

2.1. Introduction

taiga-vagrant offers a Vagrantfile which makes it easy to turn on a fully provisioned virtual machine. **taiga-vagrant** uses **taiga-scripts** for its provisioning scripts.

2.2. Important notes about using taiga-vagrant

- The installation is not 100% ready for production, it is designed for use as a demonstration.
- Does not support reprovisioning at this moment.

2.3. Dependencies

- [VirtualBox](#)
- [Vagrant](#)

2.4. Installation

```
git clone https://github.com/taigaio/taiga-vagrant.git
cd taiga-vagrant
vagrant plugin install vagrant-vbguest
vagrant up
```

After installing, the Taiga service should be up and running. You can access it by going to <http://localhost:8000> with your browser.

You can use the following credentials to login to the recently created Taiga instance:

- Username: **admin**
- Password: **123123**

You can access the machine using ssh, if you would like to make some tweaks:

```
vagrant ssh
```

3. Setup using taiga-docker

3.1. Important notes about using taiga-docker

Docker is not officially supported by the Taiga development team but a [docker script](#) has been developed by [Iván Pedrazas](#), [another docker script](#) has been developed by [Hylke Visser](#), and [yet another docker script](#) has been developed by [Benjamin Hutchins](#).