*Installatie handleiding*

Software ontwikkelstraat

*Dit is de installatie handleiding met betrekking tot het opzetten van een software ontwikkelstraat. Hiervoor is een Linux Ubuntu 12 Server opgezet en zijn daarop verschillende tools geïnstalleerd en geconfigureerd. Ook zijn er lokaal nog enkele acties ondernomen.*

**Datum:** 10-03-2014

**Groep:** S63-E

**Groepsleden:** Bas Rheiter, Ferdie Jonkers, Jelle v.d. Heijden, Remi Arts, Tim Lodder

2014

Inhoud

[SSH: 1](#_Toc384031102)

[putty: 1](#_Toc384031103)

[Netbeans: 1](#_Toc384031104)

[Jenkins: 2](#_Toc384031105)

[Jenkins systeem beheer 2](#_Toc384031106)

[Jenkins project configuratie 2](#_Toc384031107)

[GIT: 3](#_Toc384031108)

[Gitlab 4](#_Toc384031109)

[Ruby: 5](#_Toc384031110)

[PHP5 5](#_Toc384031111)

[MYSQL 5](#_Toc384031112)

[NGINX: 6](#_Toc384031113)

[Mantis: 6](#_Toc384031114)

[Sonar: 6](#_Toc384031115)

# SSH:

**Installeren:**

Apt-get install openssh-server

Apt-get install openssh-client

**Configureren:**

Nano /etc/network/interfaces

# The primary network interface

Auto eth0

Iface eth0 inet static

Address 192.168.24.150

Netmask 255.255.255.0

Gateway 192.168.24.1

Dns-nameservers 8.8.8.8

# putty:

Lokaal putty geinstalleerd zodat via ssh verbinding kan worden gemaakt met de Linux server.

# Netbeans:

Lokaal enkele plug-ins geïnstalleerd:

1. Hudson
2. Hudson UI
3. Jenkins plugin support
4. Git
5. Git toolbar
6. Maven
7. Maven utilities
8. JUnit

# Jenkins:

**Installeren:**

sudo apt-get install jenkins

## Jenkins systeem beheer

**Plugin manager:**

1. Git plugin
2. Git client plugin
3. Javadoc plugin
4. Mantis plugin
5. Sonar plugin

**Configuratie:**

* Mantis

URL: <http://192.168.24.150/mantis/>

Version: 1.2.X

User Name: administrator

Password: \*\*\*\*

* Sonar

Name: Sonar

Server URL: <http://192.168.24.150:9000/>

Sonar account login: admin

Sonar account password: \*\*\*\*

Database URL: jdbc:mysql://localhost:3306/sonar

Database login: root

Database password: \*\*\*\*

Database stuurprogramma: com.mysql.jdbc.Driver

## Jenkins project configuratie

**Mantis:**

Mantis site: <http://192.168.24.150/mantis/>

Project: MagazijnApplicatie

Category: -

**Broncodebeheer (SCM):**

Git

Repository URL: <http://192.168.24.150/root/magazijnapplicatie.git>

Credentials: Jenkins/\*\*\*\*\*\*

Branches to build: \*/Master

Repository browser: gitlab

URL: <http://192.168.24.150/>

Version: 1.8

**Acties na bouwpoging:**

* File a ticket to mantis

Build status: failure

* Publiceer javadoc

Retain Javadoc for each successful build.

* Publiceer rapport van de JUnit-testresultaten

Test report XMLs: target/surefire-reports/\*.xml

* Update relevant Mantis issues

Default

* Verzamel testresultaten van de onderliggende projecten

Automatically aggregate all downstream tests.

* Git publisher

Default

* Sonar

Default

# GIT:

**Installeren packages:**

apt-get update -y

apt-get upgrade -y

apt-get install -y build-essential zlib1g-dev libyaml-dev libssl-dev libgdbm-dev libreadline-dev libncurses5-dev libffi-dev curl openssh-server redis-server checkinstall libxml2-dev libxslt-dev libcurl4-openssl-dev libicu-dev logrotate git-core postfix

**Lokaal:**

Git downloaden op pc en daarna via git bash de folder ophalen met het volgende commando:

git clone http://192.168.24.150/root/magazijnapplicatie.git

## Gitlab

**Installeren:**

cd /home/git

sudo -u git -H git clone https://gitlab.com/gitlab-org/gitlab-ce.git -b 6-5-stable gitlab

cd /home/git/gitlab

sudo -u git -H cp config/gitlab.yml.example config/gitlab.yml

sudo -u git -H editor config/gitlab.yml

# “localhost” vervangen door “192.168.24.130”

sudo chown -R git log/

sudo chown -R git tmp/

sudo chmod -R u+rwX log/

sudo chmod -R u+rwX tmp/

sudo -u git -H mkdir /home/git/gitlab-satellites

sudo -u git -H mkdir tmp/pids/

sudo -u git -H mkdir tmp/sockets/

sudo chmod -R u+rwX tmp/pids/

sudo chmod -R u+rwX tmp/sockets/

sudo -u git -H mkdir public/uploads

sudo chmod -R u+rwX public/uploads

sudo -u git -H cp config/unicorn.rb.example config/unicorn.rb

sudo -u git -H cp config/initializers/rack\_attack.rb.example config/initializers/rack\_attack.rb

sudo -u git -H git config --global user.name "GitLab"

sudo -u git -H git config --global user.email "gitlab@localhost"

sudo -u git -H git config --global core.autocrlf input

sudo -u git cp config/database.yml.mysql config/database.yml

sudo -u git -H editor config/database.yml #secure password vervangen door sei

sudo -u git -H chmod o-rwx config/database.yml

cd /home/git/gitlab

sudo -u git -H bundle install --deployment --without development test postgres aws

sudo -u git -H bundle exec rake gitlab:setup RAILS\_ENV=production

# typ nu “yes”

sudo cp lib/support/init.d/gitlab /etc/init.d/gitlab

sudo cp lib/support/init.d/gitlab.default.example /etc/default/gitlab

sudo update-rc.d gitlab defaults 21

sudo cp lib/support/logrotate/gitlab /etc/logrotate.d/gitlab

sudo -u git -H bundle exec rake gitlab:env:info RAILS\_ENV=production

sudo service gitlab start

sudo -u git -H bundle exec rake assets:precompile RAILS\_ENV=production

**User aanmaken:**

sudo adduser --disabled-login --gecos 'GitLab' gitsudo gem install bundler --no-ri --no-rdoc

**Shell:**

cd /home/git

sudo -u git -H git clone https://gitlab.com/gitlab-org/gitlab-shell.git -b 1.8.0

cd gitlab-shell

sudo -u git -H cp config.yml.example config.yml

sudo -u git -H ./bin/install

**Database:**

cd /home/git

sudo -u git -H git clone https://gitlab.com/gitlab-org/gitlab-shell.git -b 1.8.0

cd gitlab-shell

sudo -u git -H cp config.yml.example config.yml

sudo -u git -H ./bin/install

# Ruby:

**Installeren:**

sudo apt-get remove ruby1.8

mkdir /tmp/ruby && cd /tmp/ruby

curl --progress ftp://ftp.ruby-lang.org/pub/ruby/2.0/ruby-2.0.0-p353.tar.gz | tar xz

cd ruby-2.0.0-p353

./configure --disable-install-rdoc

make

sudo make install

sudo gem install bundler --no-ri --no-rdoc

# PHP5

Sudo apt-get install php5

# MYSQL

Sudo apt-get install mysql-server

Sudo apt-get install php5-mysql

# NGINX:

**Installeren:**

sudo apt-get install -y nginx

sudo cp lib/support/nginx/gitlab /etc/nginx/sites-available/gitlab

sudo ln -s /etc/nginx/sites-available/gitlab /etc/nginx/sites-enabled/gitlab

# Mantis:

**Installeren:**

sudo apt-get install mantis

sudo nano /etc/nginx/sites-available/gitlab

location /mantis {

try\_files $uri $uri/index.php;

}

# Sonar:

**Installeren:**

MySQL --password test

Create database sonar

sudo apt-get install sonar

**Configureren**:  
cd /opt/sonar/conf

Nano sonar.properties

Sonar.jdbc.username veranderen naar root.

Sonar.jdbc.password veranderen naar \*\*\*\*

Uncommand sonar.jdbc.url onder MySQL5

Uncommand sonar.web.port = 9000

Service sonar restart