Gitlab安装

1. 将yum源设置为阿里云开源镜像

cd /etc/yum.repos.d

wget -O CentOS-Base.repo <http://mirrors.aliyun.com/repo/Centos-6.repo>

sudo rpm -Uvh http://dl.fedoraproject.org/pub/epel/6/x86\_64/epel-release-6-8.noarch.rpm

1. 必要软件包

yum -y install libicu-devel patch gcc-c++ readline-devel zlib-devel libffi-devel openssl-devel make autoconf automake libtool bison libxml2-devel libxslt-devel libyaml-devel zlib-devel openssl-devel cpio expat-devel gettext-devel curl-devel perl-ExtUtils-CBuilder perl-ExtUtils-MakeMaker

1. 安装Git

wget -O git-src.zip <https://github.com/git/git/archive/master.zip>

unzip git-src.zip

cd git-src

make prefix=/usr/local all

make prefix=/usr/local install

ln -fs /usr/local/bin/git\* /usr/bin/

4.

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

curl --progress ftp://ftp.ruby-lang.org/pub/ruby/ruby-2.1.5.tar.gz | tar xz

cd ruby-2.1.5

./configure --disable-install-rdoc

make && make install

ln -s /usr/local/bin/ruby /usr/bin/ruby

ln -s /usr/local/bin/gem /usr/bin/gem

ln -s /usr/local/bin/bundle /usr/bin/bundle

设置ruby gem源为淘宝

gem sources --add https://ruby.taobao.org/ --remove <https://rubygems.org/>

gem sources -l

\*\*\* CURRENT SOURCES \*\*\*

<https://ruby.taobao.org>

gem install rails

bundle config mirror.https://rubygems.org https://ruby.taobao.org

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

4.安装MySQL及初始化GitLab库

$ yum install mysql mysql-devel mysql-server -y

$ /etc/init.d/mysqld start

$ chkconfig mysqld on

// 登录mysql创建gitab的帐号和数据库

mysql> CREATE USER 'gitlab'@'localhost' IDENTIFIED BY 'gitlab';

mysql> CREATE DATABASE IF NOT EXISTS `gitlabhq\_production` DEFAULT CHARACTER SET `utf8` COLLATE `utf8\_unicode\_ci`;

mysql> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER ON `gitlabhq\_production`.\* TO 'gitlab'@'localhost';

//测试是否可以用git帐号登录数据库

sudo -u git -H mysql -u gitlab -p -D gitlabhq\_production

5.安装redis

wget <http://dl.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm>

rpm -ivh epel-release-6-8.noarch.rpm

yum -y install redis

/etc/init.d/redis start

chkconfig redis on

6. 添加git帐号并允许sudo

useradd --comment 'GitLab' git

echo "git ALL=(ALL) NOPASSWD: ALL" >>/etc/sudoers

7. 安装GitLab

mkdir /home/git

sudo -u git -H git clone https://gitlab.com/gitlab-org/gitlab-ce.git -b 7-8-stable gitlab

cd /home/git/gitlab

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

// 编辑git路径, gitlab的host:port

$ vim config/gitlab.yml

// bin\_path: /usr/local/bin/git

// host: localhost

// port: 80

// 给文件夹添加相应的权限

chown -R git log/

chown -R git tmp/

chmod -R u+rwX log/

chmod -R u+rwX tmp/

// 创建必要的文件夹，以及复制配置文件

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

// 配置数据库连接信息

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

sudo -u git -H vim config/database.yml

vim config/database.yml

// production:

// username: gitlab

// password: "gitlab"

8. 安装gitshell

cd /home/git

sudo -u git -H git clone [https://gitlab.com/gitlab-org/gitlab-shell.git -b v2.6](https://gitlab.com/gitlab-org/gitlab-shell.git%20-b%20v2.6).0

cd gitlab-shell/

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

// 编辑配置文件, 设置gitlab\_url, redis-cli, log-level...

vim config.yml

// gitlab\_url: "http://localhost/"

// /usr/bin/redis-cli

// 安装git-shell

sudo -u git -H ./bin/install

9. 安装gems

cd /home/git/gitlab

vi Gemfile

把#source "https://rubygems.org"

改成 source <https://ruby.taobao.org>

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

如果报错

gem install coffee-script-source -v '1.6.3'

gem install rugged -v '0.21.2'

如果还是报错

sudo yum -y install cmake

gem install mysql2 -v '0.3.16'

gem install redis-activesupport -v '4.0.0'

10. 初始化数据库(创建GitLab相关表)

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

11.安装启动文件以及日志切割文件

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

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

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

12. 设置git帐号信息

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

sudo -u git -H git config --global user.email "1073707152@qq.com"

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

13 .安装nginx

安装Nginx

$ yum -y install nginx

$ vim /etc/nginx/nginx.conf

user root git;

worker\_processes 2;

pid /var/run/nginx.pid;

events {

worker\_connections 1024;

}

http {

include /etc/nginx/mime.types;

default\_type application/octet-stream;

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" "$http\_x\_forwarded\_for"';

# GITLAB

# Maintainer: @randx

# App Version: 5.0

upstream gitlab {

server unix:/home/git/gitlab/tmp/sockets/gitlab.socket;

}

server {

listen \*:80 default\_server; # e.g., listen 192.168.1.1:80; In most cases \*:80 is a good idea

server\_name YOUR\_SERVER\_FQDN; # e.g., server\_name source.example.com;

server\_tokens off; # don't show the version number, a security best practice

root /home/git/gitlab/public;

# Set value of client\_max\_body\_size to at least the value of git.max\_size in gitlab.yml

client\_max\_body\_size 5m;

# individual nginx logs for this gitlab vhost

access\_log /var/log/nginx/gitlab\_access.log;

error\_log /var/log/nginx/gitlab\_error.log;

location / {

# serve static files from defined root folder;.

# @gitlab is a named location for the upstream fallback, see below

try\_files $uri $uri/index.html $uri.html @gitlab;

}

# if a file, which is not found in the root folder is requested,

# then the proxy pass the request to the upsteam (gitlab unicorn)

location @gitlab {

proxy\_read\_timeout 300; # https://github.com/gitlabhq/gitlabhq/issues/694

proxy\_connect\_timeout 300; # https://github.com/gitlabhq/gitlabhq/issues/694

proxy\_redirect off;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_set\_header Host $http\_host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_pass http://gitlab;

}

}

}

更改权限，启动nginx

$ nginx -t

$ chown -R git:git /var/lib/nginx/

$ /etc/init.d/nginx start

14. 检测当前环境

cd /home/git/gitlab

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

拉取gitlab静态资源文件

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

更改数据库配置

cd /home/git/gitlab/config

vi database.yml.mysql

production:

adapter: mysql2

encoding: utf8

collation: utf8\_general\_ci

reconnect: false

database: gitlabhq\_production

pool: 10

username: gitlab

password: "gtilab"

host: localhost

socket: /tmp/mysql.sock

15.启动

/etc/init.d/gitlab start