

# GIT

## Gogs

Go语言开发的Git服务器

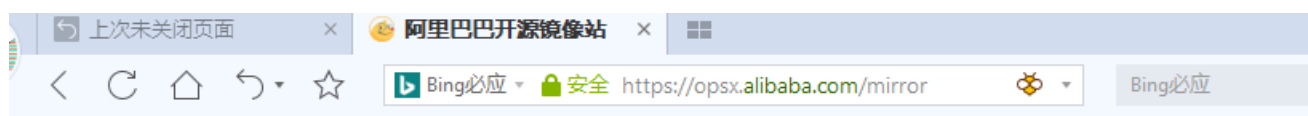
<https://gogs.io/>

## 软件依赖

本次安装使用CentOS7

yum库

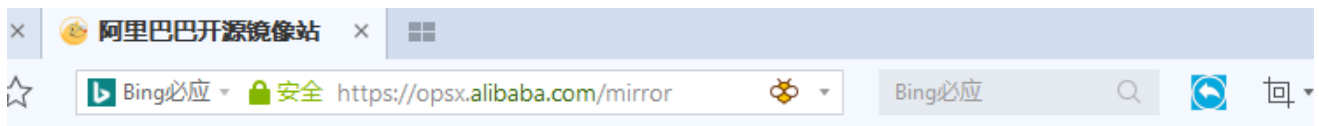
访问阿里镜像站<http://mirrors.aliyun.com>，找到centos，点击右边“帮助”



OPSX   开放 共享 连接					首
名称	标签	更新时间	同步状态	操作	
centos	系统	2019-02-25 01:57:16	● 成功	帮助	

下载之前，可以移除原有repo文件。

找到对应CentOS对应版本的repo文件下载。



centos

## CentOS

### 1、备份

```
mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup
```

### 2、下载新的CentOS-Base.repo 到/etc/yum.repos.d/

CentOS 5

```
wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-5.repo
```

或者

```
curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-5.repo
```

CentOS 6

```
wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo
```

或者

```
curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-6.repo
```

CentOS 7

```
wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo
```

或者

```
curl -o /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo
```

## Git 安装

```
# yum install git -y
```

## MariaDB 安装

```
# yum list | grep mariadb
```

mariadb-libs.x86_64	1:5.5.60-1.el7_5	@anaconda
mariadb.x86_64	1:5.5.60-1.el7_5	base
mariadb-bench.x86_64	1:5.5.60-1.el7_5	base
mariadb-devel.i686	1:5.5.60-1.el7_5	base
mariadb-devel.x86_64	1:5.5.60-1.el7_5	base
mariadb-embedded.i686	1:5.5.60-1.el7_5	base
mariadb-embedded.x86_64	1:5.5.60-1.el7_5	base
mariadb-embedded-devel.i686	1:5.5.60-1.el7_5	base
mariadb-embedded-devel.x86_64	1:5.5.60-1.el7_5	base
mariadb-libs.i686	1:5.5.60-1.el7_5	base
mariadb-server.x86_64	1:5.5.60-1.el7_5	base
mariadb-test.x86_64	1:5.5.60-1.el7_5	base

安装mariadb 服务，会自动安装mairadb

```
# yum install mariadb-server

# systemctl start mariadb.service
# ss -tanl
State          Recv-Q Send-Q Local Address:Port          Peer Address:Port
LISTEN         0      50    *:3306                    *.*

开机启动
# systemctl enable mariadb.service
# mysql_secure_installation

数据库密码登录
# mysql -u root -p
mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| performance_schema |
+-----+
3 rows in set (0.00 sec)
```

## 安装

Gogs服务以git用户运行

```
# useradd git
# su - git

$ cd gogs
```

下载gogs安装程序gogs0.11.4\_amd64.tar.gz，上传到Centos解压，生产gogs目录，里面就是gogs所有文件。

初始化数据库

```
$ mysql -uroot -p < scripts/mysql.sql # 其实就是创建了一个gogs的库

# 为gogs库创建mysql用户gogs，并授权
mysql> grant all on gogs.* to 'gogs'@'%' identified by 'gogs';
mysql> flush privileges;
```

## 配置

参考官方文档 [https://gogs.io/docs/advanced/configuration\\_cheat\\_sheet.html](https://gogs.io/docs/advanced/configuration_cheat_sheet.html)

概述

名称	描述
APP_NAME	应用名称，可以改成您的组织或公司名称
RUN_USER	运行应用的用户名称，我们建议您使用 git，但如果您在个人计算机上运行 Gogs，请修改为您的系统用户名称。如果没有正确设置这个值，很可能导致您的应用崩溃

#### 服务器 (server)

名称	描述
HTTP_ADDR	应用 HTTP 监听地址
HTTP_PORT	应用 HTTP 监听端口号
DOMAIN	服务器域名，会影响 SSH clone 地址

ROOT\_URL = %(PROTOCOL)s://%(DOMAIN)s:%(HTTP\_PORT)s/

#### 仓库 (repository)

名称	描述
ROOT	用户仓库存储根目录，必须为绝对路径，默认为 <code>~/&lt;user name&gt;/gogs-repositories</code>

#### 数据库 (database)

名称	描述
DB_TYPE	数据库类型，可以是 mysql、postgres、mssql 或 sqlite3
HOST	数据库主机地址与端口
NAME	数据库名称
USER	数据库用户名
PASSWD	数据库用户密码

#### 安全 (security)

名称	描述
INSTALL_LOCK	用于指示是否允许访问安装页面（该页面可以设置管理员帐号，因此该选项非常重要）
SECRET_KEY	全局的加密密钥，务必修改该值以确保您的服务器安全（会在每次安装时自动生成随机字符串）

#### 配置文件

在gogs目录下建立custom/conf/app.ini配置文件

```
$ mkdir -p custom/conf
$ cd custom/conf
$ touch app.ini
```

custom/conf/app.ini内容如下

```
APP_NAME = magedu
RUN_USER = git
RUN_MODE = dev
[server]
HTTP_ADDR = 0.0.0.0
HTTP_PORT = 3000

[database]
DB_TYPE = mysql
HOST = 127.0.0.1:3306
NAME = gogs
USER = gogs
PASSWD = gogs

[security]
INSTALL_LOCK = false
SECRET_KEY = www.magedu.com.(python).GIT:gogs
```

注意app.ini的权限、属主、属组。

## 启动gogs

### 1、使用gogs命令启动

```
$ ./gogs web
```

但是这样用不方便，使用scripts目录下的服务脚本

如果不能访问，请查看防火墙

```
# systemctl status firewalld
# systemctl stop firewalld
```

**在gogs目录下建一个log目录，否则某些gogs版本可能启动不了**

### 2、服务启动

```
# cp /home/git/gogs/scripts/systemd/gogs.service /lib/systemd/system/  
# systemctl start gogs  
# systemctl enable gogs  
# systemctl enable mariadb  
# systemctl stop firewalld  
# systemctl disable firewalld
```

首次登陆

<http://192.168.142.135:3000/install>

## 首次运行安装程序

如果您正在使用 Docker 容器运行 Gogs，请务必先仔细阅读 [官方文档](#) 后再对本页面进行填写。

### 数据库设置

Gogs 要求安装 MySQL、PostgreSQL、SQLite3、MSSQL 或 TiDB。

数据库类型 \* MySQL

数据库主机 \* 127.0.0.1:3306

数据库用户 \* gogs

数据库用户密码 \* ....

数据库名称 \* gogs

如果您使用 MySQL，请使用 INNODB 引擎以及 utf8\_general\_ci 字符集。

### 应用基本设置

应用名称 \* magedu

快用狂拽酷炫的组织名称闪瞎我们！

仓库根目录 \* /home/git/gogs-repositories

所有 Git 远程仓库都将被存放于该目录。

运行系统用户 \* git

该用户必须具有对仓库根目录和运行 Gogs 的操作权限。

域名 \* 192.168.142.140

该设置影响 SSH 克隆地址。

SSH 端口号 \* 22

SSH 服务器的监听端口号，留空表示禁用 SSH 功能。

☐ 使用内置 SSH 服务器

HTTP 端口号 \* 3000

应用监听的端口号

应用 URL \* http://192.168.142.140:3000/

该设置影响 HTTP/HTTPS 克隆地址和一些邮箱中的链接。

日志路径 \* /home/git/gogs/log

存放日志文件的目录

☐ 启用控制台模式

然后点击 [立即安装](#)

需要注册第一个用户，这个用户直接成为管理员账户。

## 登录

您好！我们很高兴您选择使用 Gogs，祝您使用愉快，代码从此无 BUG！

用户名或邮箱 \*

密码 \*

☐ 记住登录

登录

[忘记密码？](#)

[还没帐户？](#)

[马上注册。](#)

由于一个用户都没有，所以注册第一个用户自动成为管理员

## 注册

用户名 \*

gitadmin

邮箱 \*

admin@magedu.com

密码 \*

.....

确认密码 \*

.....



验证码 \*

775704

创建帐户

[已经注册？立即登录！](#)

使用第一个用户登录，由于此用户同时是管理员，可以进行管理。



[控制面板](#)[工单管理](#)[合并请求](#)[探索](#)

+



已登录用户 GITADMIN

[个人信息](#)[用户设置](#)[帮助](#)[管理面板](#)[退出](#)[执行](#)[执行](#)[执行](#)[执行](#)[执行](#)[执行](#)

管理面板

控制面板

用户管理

组织管理

仓库管理

授权认证管理

应用配置管理

系统提示管理

应用监控面板

应用统计数据

Gogs 数据库统计: 1 位用户, 0 个组织, 0 个公钥, 0 个仓库, 0 个仓库关注, 0 个赞, 0 次行为, 0 条权限次评论, 0 个社交帐号, 0 个用户关注, 0 个镜像, 0 个版本发布, 0 个登录源, 0 个 Web 钩子, 0 个里程碑子任务, 0 个团队, 0 个更新任务, 0 个附件。

管理员操作

删除所有未激活帐户

删除所有仓库存档

删除所有丢失 Git 文件的仓库记录

对仓库进行垃圾回收

重新生成 '.ssh/authorized\_keys' 文件 (警告: 不是 Gogs 的密钥也会被删除)

重新同步所有仓库的 pre-receive、update 和 post-receive 钩子

重新初始化所有丢失 Git 文件的仓库

系统监视状态

## 使用

创建第二个账户, 普通用户, 创建后登录

### 注册

用户名 \*

my

邮箱 \*

my@magedu.com

密码 \*

..

确认密码 \*

..



验证码 \*

721328

[创建帐户](#)[已经注册? 立即登录!](#)





仓库

组织

镜像

我的仓库 0

创建新的仓库



参与协作的仓库

## 创建新的仓库

拥有者 \*



my

仓库名称 \*

test

伟大的仓库名称一般都较短、令人深刻并且独一无二的。

可见性



该仓库为 私有的

仓库描述

测试用

.gitignore

选择 .gitignore 模板

授权许可

请选择授权许可文件

自述文档 ?

Default



使用选定的文件和模板初始化仓库

创建仓库

取消



my / test

取消关注

1

点赞

0

文件



工单管理 0



Wiki



仓库设置

### 快速帮助

克隆当前仓库 不知道如何操作？访问 [此处](#) 查看帮助！

HTTP

SSH

http://192.168.142.135:3000/my/test.git



### 从命令行创建一个新的仓库

```
touch README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin http://192.168.142.135:3000/my/test.git
git push -u origin master
```

### 从命令行推送已经创建的仓库

```
git remote add origin http://192.168.142.135:3000/my/test.git
git push -u origin master
```

好了，这样一个自己搭建的Git私服已经准备好了，可以使用客户端连接推送文件了。

## gogs问题解决

### 1、打开网页慢

日志中可以看到 AvatarLink.LibravatarService.FromEmail [my@magedu.com]: lookup \_avatars-sec.\_tcp.magedu.com on 192.168.142.2:53: read udp 192.168.142.135:40255->192.168.142.2:53: i/o timeout

访问不到gravatar.com的头像服务，解决办法就是禁用gravatar服务，使用本地头像  
打开配置文件，修改如下

```
[picture]
DISABLE_GRAVATAR          = true
ENABLE_FEDERATED_AVATAR  = true
```

重新加载配置即可

也可以在访问install的时候，就禁用Gravatar服务

## 可选设置

---

### ▶ 邮件服务设置

### ▼ 服务器和其它服务设置

☐ 启用离线模式

☒ 禁用 Gravatar 服务

☐ 启用 Federated Avatars 查找

☐ 禁止用户自主注册

☒ 启用验证码服务

☐ 启用登录访问限制

### ▼ 管理员帐号设置