**版本管理工具Git-客户端与服务器端的安装与配置**

# 一、版本管理工具简介

在安装之前先简单介绍一下版本管理工具，版本管理工具有集中式管理工具和分布式管理工具，其中集中式管理工具有CVS、SVN，分布式管理工具有git。git是2005年开发的，不是一种新工具，只是国内使用的比较晚。

集中式管理工具和分布式管理工具的主要区别在于：

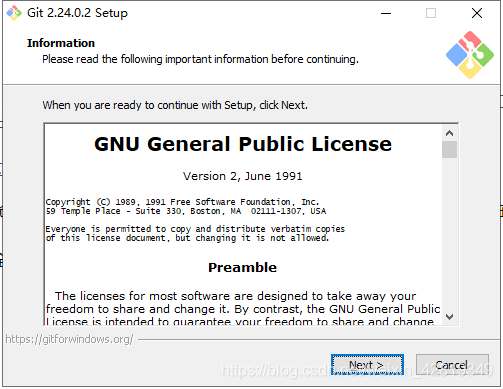
集中式管理工具有一个统一的中央仓库，在没有网络的情况下不能同步。分布式管理工具的仓库在本地，在没有网络的情况下也可以同步。

集中式管理工具每个版本的代码是全量，分布式管理工具的代码是增量。

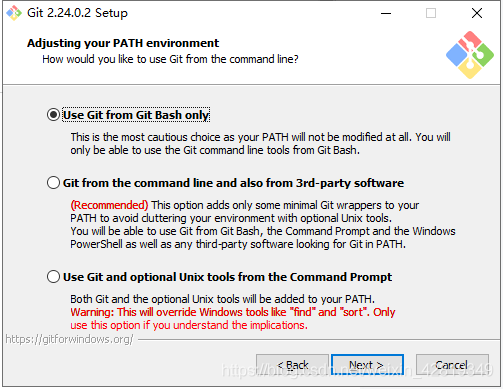
# 二、客户端的安装

1、git下载地址：msysgit.github.io

2、下载后双击安装



一直点下一步，直到这个界面，选择第一项。



之后一直点下一步，进行安装。

3、配置环境变量

PATH路径：C:\Program Files\Git\bin

4、配置git的用户名和邮箱

屏幕右键，选择Git Bash Here



创建用户名和邮箱，其中邮箱就是密码。

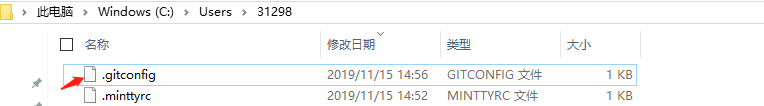
命令：

git config --global user.name "zbgit"

git config --global user.email "zbgit@qq.com"



查看命令是否执行成功，打开当前登录用户用录下的.gitconfig文件，文件内容为刚创建的用户名和邮箱



# 三、搭建git服务器(远程仓库)

在git统一的拖管网站进行注册

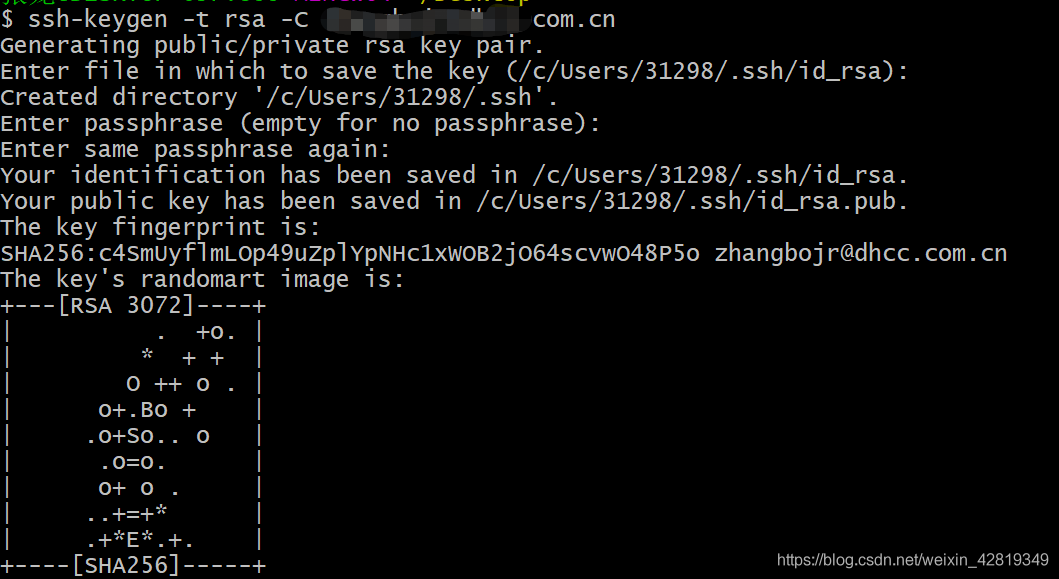
https://github.com/

通过配置SSH进行免密钥登录，目的是本地机器跟远程服务器可以免密钥

先在本地配置SSH

在屏幕空白处点击右键，选择GIT Bash Here,输入命令：

ssh-keygen -t rsa -C zbgit@qq.com

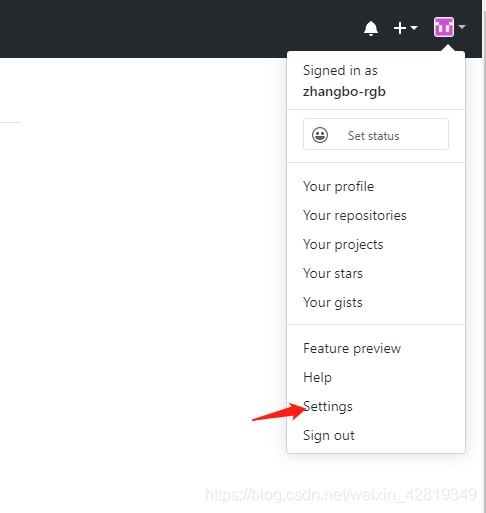


查看是否配置成功

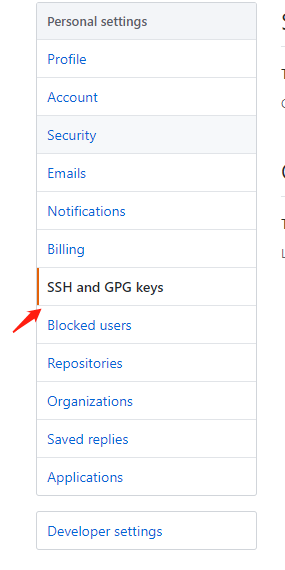
在C:\Users/当前用户/.ssh下查看是否有两个文件，分别为id\_rsa、id\_rsa.pub。

将ssh发送给远程

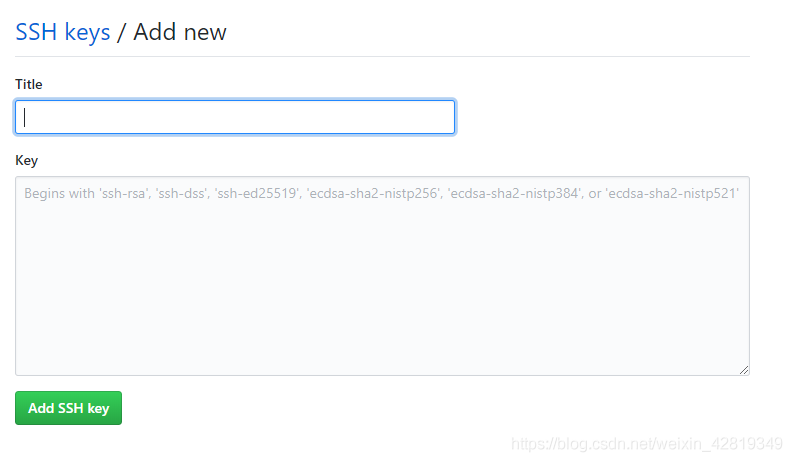
打开github.com,登录后，选择Settings.



选择SSH and GPG keys

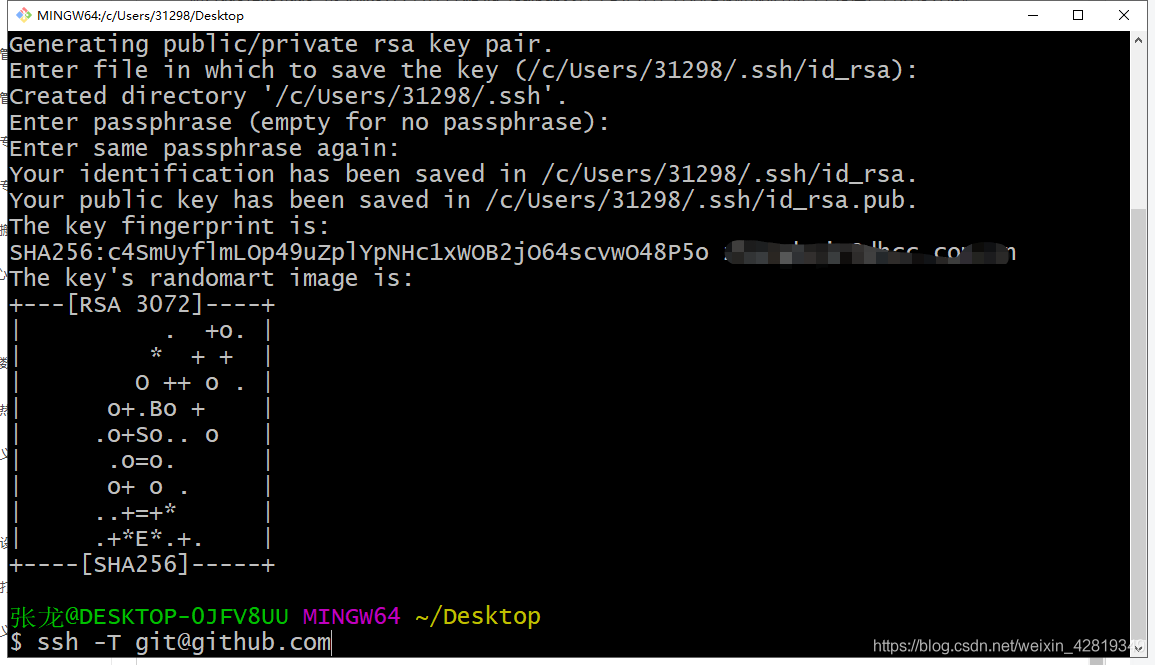


输入title和key,title可随意,key为本地生成的id\_rsa.pub文件中的内容，注意复制粘贴后将回车去掉，否则会失败。



测试连通性：

ssh -T git@github.com



如果本地和远程通信成功，可以在C:\Users\当前用户\.ssh下能查到known\_hosts文件。

# 四、将本地项目修改为git项目

在项目的根目录，右键选择Git Bash Here.输入命令：

git init

# 五、在github上新建项目

登录gibhub网站，点击Your Repositories,Repository Name可随意，属性选择Public.

建立后会生成一个唯一标识，https和ssh都可以：https://github.com/zhangbo-rgb/gitremote.git

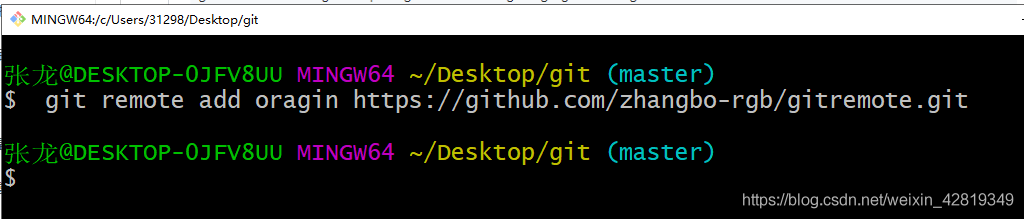


# 六、将本地项目与github上的项目关联

在Git Bash Here中执行命令

git remote add origin https://github.com/zhangbo-rgb/gitremote.git

下图中的oragin写错了，应该是origin



这样我们就完成通过SSH将本地与远程服务器进行了关联，并且将本地的项目与远程项目进行了关联。

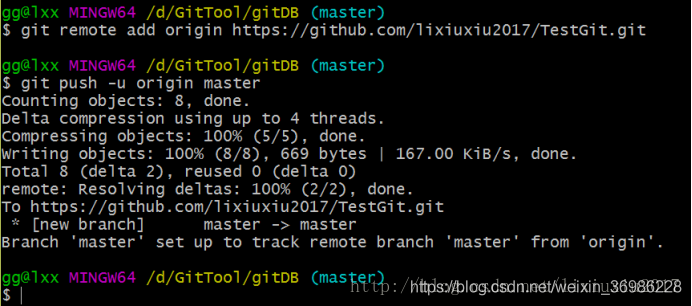
# 七、将本地Git仓库中的文件同步推送到GitHub中

在本地Git仓库目录下运行如下命令：

在下面的命令之前先输入:git init，否则易报not a git repository.

git remote add origin https://github.com/lixiuxiu2017/TestGit.git

git push -u origin master（由于新建的GitHub仓库是空的，所以第一次推送master分支时需要加-u参数，以后再推送就不用加了）



推送过程中可能需要输入github的用户名和密码，推送成功后，可以立刻在github页面中看到远程库的内容已经和本地一模一样了。

从现在起，只要本地作了提交，就可以通过如下命令：git push origin master把本地master分支的最新修改推送到github上了，现在你就拥有了真正的分布式版本库了