本文主要是Git和Github远程同步过程出现的问题和解决方

以下是全文需要注意的几个事物:

1. Github注册账号的邮箱地址
2. 本地电脑安装git的公钥

介绍下git和github工作原理:

1. Git是本地工作区的一个repository(也叫作版本库),

它主要用来跟踪某个工作目录下的所有文件修改状态

1. 通过 git init 创建某个工作区的版本库,里面存放修改信息
2. 通过 git add . / git add file 将工作区文件放入一个叫做暂存区的地方，

暂存区等待下次提交到版本库中（可以认为是缓冲区）

1. 通过git commit –m “附加信息” 将暂存区的所有数据全部提交给版本库

这样我们完成了一次项目文件的版本更新操作

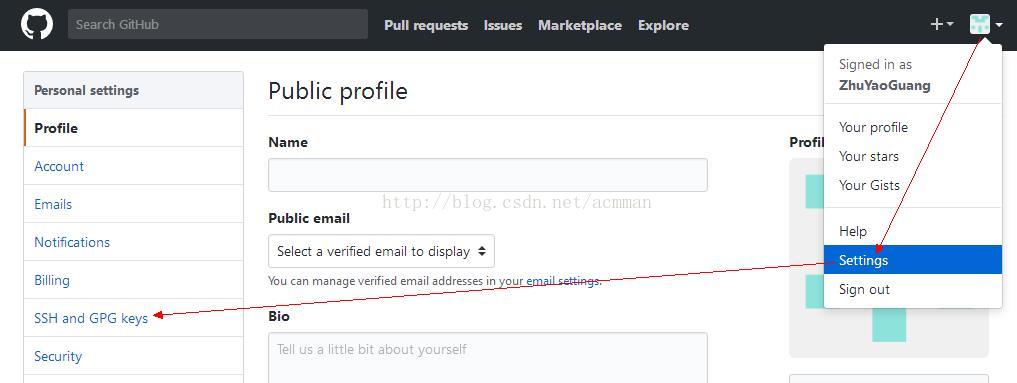
1. Github是远程仓库，类似于本地仓库，不同点在于，远程仓库便利于开源项目的开发，可以非常多的人参与项目开发，同时也是数据备份的不二选择
2. 先通过注册Github账号，需要填写我们的邮箱地址
3. 登录后，就可以自己创建一个空的远程仓库。
4. 本地下载的git仓库需要设置本地的rsa公钥，

ssh-keygen -t rsa -C [your\_email@youremail.com](mailto:your_email@youremail.com)

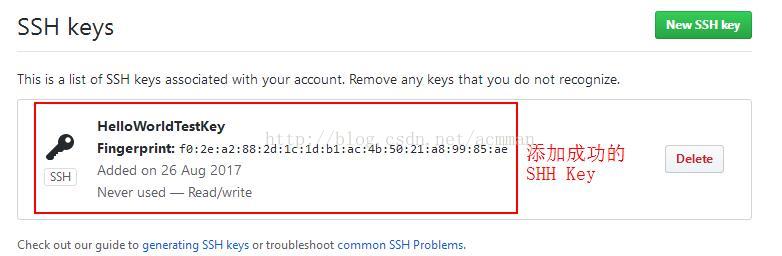
以上命令来设置本机的ssh 公钥，而email就是github账户邮箱

只有使用远程仓库账号的邮箱来设置本机秘钥，才能将本机的数据提交到远程仓库上。

1. 找到.ssh文件 并将id\_rsa.pub 文件中的key复制，在github中打开

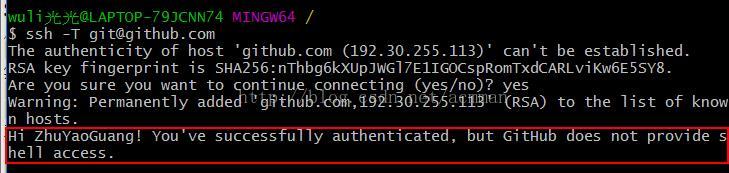


将复制的key 添加到SSH中，我们就完成了远程仓库和本地仓库的公钥秘钥配置

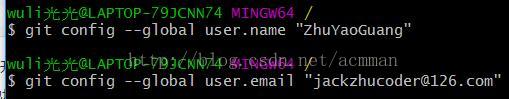


1. 在本地的git-bash中验证

ssh -T [git@github.com](mailto:git@github.com)



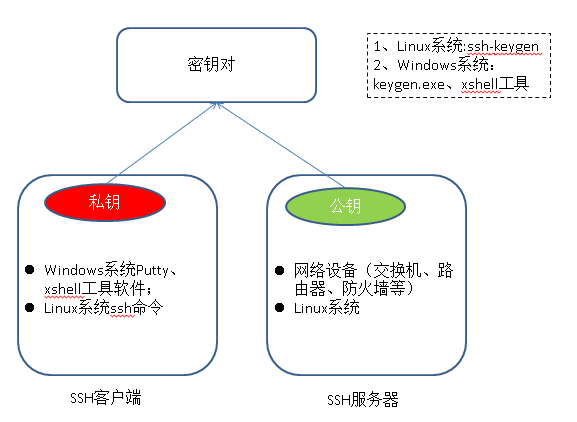
1. 本地的git公钥生成后，还需要配置username和useremail



这样每次commit提交的数据信息中都会记录哪些数据是你提交的，相当于对数据做了一个实名认证。

总结：

1. 本地git 的user.name 和 user.email 用于辨别提交的数据版本来自谁
2. 本机通过github的邮箱地址生成的公钥秘钥对的工作



本地保存秘钥，在github服务端输入公钥，当我们向服务器提交数据时，会做公钥秘钥验证。

配置过程可能遇到的问题:

$ git push -u origin master

To git@github.com:yangchao0718/cocos2d.git

 ! [rejected]        master -> master (non-fast-forward)

error: failed to push some refs to 'git@github.com:yangchao0718/cocos2d.git

hint: Updates were rejected because the tip of your current branch is behin

hint: its remote counterpart. Integrate the remote changes (e.g.

hint: 'git pull ...') before pushing again.

发现不能给远程仓库push数据,问题是远程的一些数据不在本地,需要做同步

git pull --rebase origin master

pull 相当于 将远程数据获取,并和本地数据合并(注意相同的文件可能存在合并失败)