

初次安装git配置用户名和邮箱

初次安装git需要配置用户名和邮箱，否则git会提示：please tell me who you are.

```
*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"
```

你需要运行命令来配置你的用户名和邮箱：

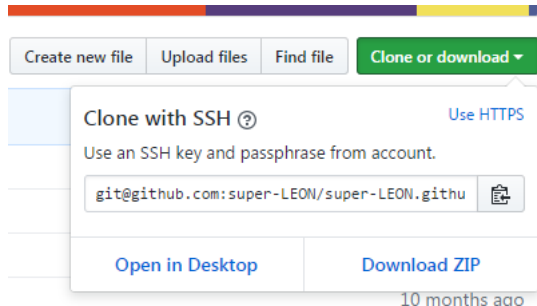
```
$ git config --global user.name "superGG1990"
$ git config --global user.email "superGG1990@163.com"
```

注意：（引号内请输入你自己设置的名字，和你自己的邮箱）此用户名和邮箱是git提交代码时用来显示你身份和联系方式的，并不是github用户名和邮箱

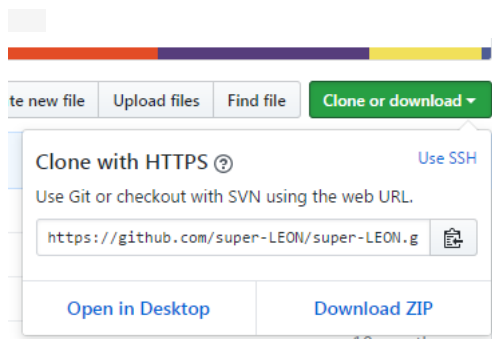
git使用ssh密钥

git支持https和git两种传输协议，github分享链接时会有两种协议可选：

git协议链接图例：↓



https协议链接图例：↓



git使用https协议，每次pull, push都会提示要输入密码，使用git协议，然后使用ssh密钥，这样免去每次都输密码的麻烦

初次使用git的用户要使用git协议大概需要三个步骤：

一、生成密钥对

二、设置远程仓库（本文以github为例）上的公钥

三、把git的 remote url 修改为git协议（以上两个步骤初次设置过以后，以后使用都不需要再次设置，此步骤视以后项目的remote url而定，如果以后其他项目的协议为https则需要此步骤）

一、生成密钥对

大多数 Git 服务器都会选择使用 SSH 公钥来进行授权。系统中的每个用户都必须提供一个公钥用于授权，没有的话就要生成一个。生成公钥的过程在所有操作系统上都差不多。首先你要确认一下本机是否已经有一个公钥。

SSH 公钥默认储存在账户的主目录下的 ~/.ssh 目录。进去看看：

```
$ cd ~/.ssh
$ ls
authorized_keys2  id_dsa      known_hosts  config      id_dsa.pub
```

看一下有没有id_rsa和id_rsa.pub(或者是id_dsa和id_dsa.pub之类成对的文件)，有 .pub 后缀的文件就是公钥，另一个文件则是密钥。

假如没有这些文件，甚至连 .ssh 目录都没有，可以用 ssh-keygen 来创建。该程序在 Linux/Mac 系统上由 SSH 包提供，而在 Windows 上则包含在 MSysGit 包里：

```
$ ssh-keygen -t rsa -C "your_email@youremail.com"
```

Creates a new ssh key using the provided email # Generating public/private rsa key pair.

Enter file in which to save the key (/home/you/.ssh/id_rsa):

直接按Enter就行。然后，会提示你输入密码，如下(建议输一个，安全一点，当然不输也行，应该不会有人闲的无聊冒充你去修改你的代码)：

Enter same passphrase again: [Type passphrase again]

完了之后，大概是这样：

Your public key has been saved in /home/you/.ssh/id_rsa.pub.

The key fingerprint is: # 01:0f:f4:3b:ca:85:d6:17:a1:7d:f0:68:9d:f0:a2:db your_email@youremail.com

到此为止，你本地的密钥对就生成了。

二、添加公钥到你的远程仓库 (github)

1、查看你生成的公钥：

```
$ cat ~/.ssh/id_rsa.pub
```

```
ssh-rsa
```

```
AAAAB3NzaClyc2EAAAADAQABAAQCAQC0X6L1zLL4VHuvGb8aJH3ippTozmReSUzgnvk434aJ/v7kOdJ/MTyBlWXFCR+HAo3FXRitBqxiXlnKhXpHAZsMciLq8v  
admin@admin-PC
```

2、登陆你的github帐户。点击你的头像，然后 Settings -> 左栏点击 SSH and GPG keys -> 点击 New SSH key

3、然后你复制上面的公钥内容，粘贴进“Key”文本域内。 title域，自己随便起个名字。

4、点击 Add key。

完成以后，验证下这个key是不是正常工作：

```
$ ssh -T git@github.com
```

Attempts to ssh to github

如果，看到：

```
Hi xxx! You've successfully authenticated, but GitHub does not # provide shell access.
```

恭喜你，你的设置已经成功了。

三、修改git的remote url

使用命令 git remote -v 查看你当前的 remote url

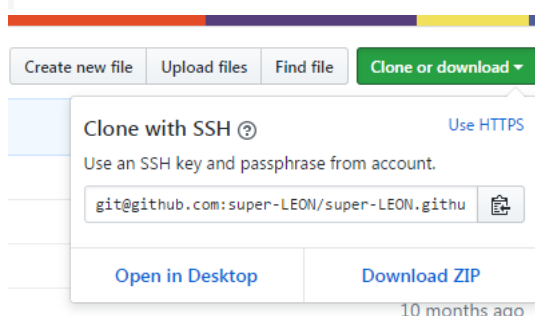
```
$ git remote -v
```

```
origin https://github.com/someaccount/someproject.git (fetch)
```

```
origin https://github.com/someaccount/someproject.git (push)
```

如果是以上的结果那么说明此项目是使用https协议进行访问的（如果地址是git开头则表示是git协议）

你可以登陆你的github，就像本文开头的图例，你在上面可以看到你的ssh协议相应的url，类似：



复制此ssh链接，然后使用命令 git remote set-url 来调整你的url。

```
git remote set-url origin git@github.com:someaccount/someproject.git
```

然后你可以再用命令 git remote -v 查看一下，url是否已经变成了ssh地址。

然后你就可以愉快的使用git fetch, git pull, git push，再也不用输入烦人的密码了