服务器端：

第一步：配置git

1.1 从yum上安装git

yum –y install git

1.2 查看git版本：安装完成后，在服务器端输入

git --version

就可以查看当前版本号码，如下图所示我的git版本是~~1.8.3.1~~

1.3 创建用户：在服务器中创建一个git专属用户

//以下为CentOS下的用户，账户密码自定义

useradd git –s /usr/bin/git-shell

passwd git

1.4 禁止该git用户使用shell登录系统（为了安全性，一般都禁止）

vi /etc/passwd

 按i进入编辑模式，在最后一行将git用户修改成以下配置

git:x:1000:1000::/home/git:/usr/bin/git-shell

按ESC退出插入模式，输入 “:wq” 保存并且退出vi模式

第二步：配置远程仓库

2.1 创建一个空仓库

我们选择在路径：cd  /home/ 下先创建一个用户目录，在用户目录下创建一个git仓库

cd /home

mkdir git

cd git

git init --bare LearnProject.git

到这里，空仓库已经创建成功 ，仓库的路径为：/home/git/LearnProject.git

2.2 为刚刚创建的用户git赋予权限

输入以下命令，为git用户赋予权限

//chown -R 用户名:组名 文件

chown -R git:git LearnProject.git

第三步：打开 RSA 认证

进入/etc/ssh目录，编辑 sshd\_config

cd /etc/ssh

vi ssh\_config

 按i进入插入模式，打开以下三个配置的注释（带#为注释）, 按ESC退出插入模式，按:wq保存

RSAAuthentication yes

PubkeyAuthentication yes

AuthorizedKeysFile .ssh/authorized\_keys

保存完成后，重启sshd服务

/etc/rc.d/init.d/sshd restart

如果运行上述命令出现以下错误：

https://img-blog.csdn.net/20180814111435740?watermark/2/text/aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L2JiY2Nra2w=/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70

可以尝试以下命令：

service sshd restart

至此，服务器端git配置完成；

客户端

第一步：配置git

1.1 安装git

在Linux上参考服务器端安装git，在Windows下教程很多，安装也傻瓜式，这里不再赘述

1.2 打开git bash创建git使用者

git config --global user.name "Tanzj"

git config --global user.email "im.tanzj@live.com"

1.3 创建与服务器通讯的密钥

在git bash中，输入以下命令，为自己的邮箱创建一个密钥

ssh-keygen -t rsa -C "im.tanzj@live.com"

一路enter下去

这时候，系统已经为我们创建了一个公钥和密钥，id\_rsa是密钥，不能告诉任何人，id\_rsa.pub是公钥，可以公开；

1.4 将本地的公钥添加到服务器上

1.4.1 通过客户端命令行将本地公钥添加进远程服务器中

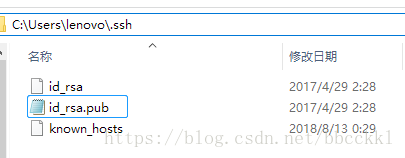
在gitbash下，输入以下命令（ip地址换成你git服务器的地址）

ssh git@服务器ip 'cat >> .ssh/authorized\_keys' < ~/.ssh/id\_rsa.pub

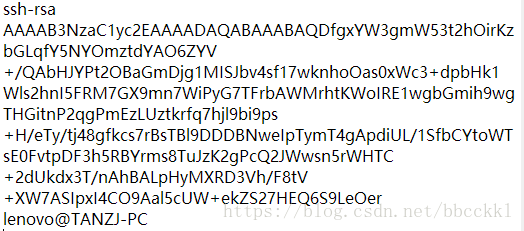
1.4.2 手动将公钥添加到服务器中

在1.3中，我们创建了一个公钥，以Windows系统为例子，我们先找到这个文件。这个文件的路径在：

C:\Users\Administrator\.ssh\id\_rsa.pub



用记事本的方式打开，我们会得到一串这样的东西



将以上Ctrl+A全选然后复制。

在服务器端，我们用vi编辑器访问authorized\_keys，将以上信息粘贴进其中

vi /home/git/.ssh/authorized\_keys

 然后:wq保存。

通过以上1.4.1和1.4.2两种方法，我们就将本地的公钥添加进服务器中了。如果发现该目录下没有.ssh文件，我们可以手动创建一个，并添加一个authorized\_keys

第二步：克隆远程仓库

将远程空仓库克隆到本地

2.1 创建本地文件夹

假设我们在D:/gitRepo/gitTest下创建一个用于本地的git仓库文件夹

cd D:

mkdir gitRepo

cd gitRepo

mkdir gitTest

2.2 克隆远程仓库

//端口在22的情况下

git clone git@172.16.162.40:/home/git/LearnProject.git

//端口在非22的情况下（如5000）

git clone ssh://git@172.16.162.40:5000/home/git/LearnProject.git

这时候，bash会提示你克隆的仓库是空的（的确是空的hhh）

至此，我们就完成了远程仓库的克隆。

第三步：push/pull项目

做完以上服务器端/客户端各两步的内容，你的本地git已经可以和git服务器进行通信了，此时我们可以进行常规的项目push和pull了。

3.1 为仓库添加内容

这里我们随便把自己的一个项目丢进去（或者随意创建一个文件亦可，空文件夹git不会识别）

3.2 提交修改

然后运行以下命令

git add .

git commit -m 初始化项目

这时候，我们就为本地创建了一次commit

3.3 push本地修改

运行以下命令

// origin 是远程仓库的别名，可以通过 git remote -v 查看

//master 是默认主分支

git push origin master

见到以上内容，便说明push成功。

3.3 git服务器远程仓库查看commit

  进入到服务器远程仓库的目录

cd /home/git/LearnProject.git

git log

看到我们的push已经生效。

至此，本文结束，如有错误，请不吝指教！

---------------------

作者：CoderTanzJ

来源：CSDN

原文：https://blog.csdn.net/bbcckkl/article/details/81634761

版权声明：本文为博主原创文章，转载请附上博文链接！