**Linux服务器GIT仓库配置和使用说明**

# 一、LINUX离线安装 Git

查看是否已安装git

|  |
| --- |
| git --version |

66服务器上 git version 1.8.3.1

本文为lunix 系统下离线安装git，如有安装过git,请先删除git相关文件

centos ,Unbantu 等平台的安装方式略有不同.

Git官网下载Git

<https://mirrors.edge.kernel.org/pub/software/scm/git/>

解压文件

|  |
| --- |
| tar -vxf /usr/local/git/git-2.9.5.tar.gz |

编译 & 安装

|  |
| --- |
| ./configure --prefix=/usr/local/git  make && make install |

环境变量配置

|  |
| --- |
| # 编辑profile文件 vim /etc/profile # 在porfile最后一行加上  export PATH=$PATH:/usr/local/git/bin # 生效配置 source /etc/profile |

验证是否安装成功

|  |
| --- |
| git --version |

创建git用户，管理git服务

|  |
| --- |
| useradd git passwd git # 创建仓库 mkdir -p data/git/gittest.git git init --bare data/git/gittest.git cd data/git/  # 将hangl.git的所有者改为git组下的git用户 chown -R git:git gittest.git/ |

客户端clone

|  |
| --- |
| git clone git@10.221.5.128:/opt/app/gitdata/gittest.git |

问题归纳

1. bash: git-receive-pack: command not found fatal: Could not read from remote repository.

解决方法：

|  |
| --- |
| ln -s /usr/local/git/bin/git-receive-pack /usr/bin/git-receive-pack |

# 二、服务器仓库搭建（LINUX, 离线）

**第1步：安装 Linux，安装 Git**

此处省略具体操作步骤。

**第2步：创建 git 用户**

在Linux系统内以管理员权限（root）执行如下命令创建Linux用户：

|  |
| --- |
| adduser git |

此命令执行过程中会提示输入新用户的 登录密码，同时自动创建用户主目录/home/git，以及执行其他必要的工作。

该用户不一定非叫git，任意合法名称都行，例如 liigogit 等等，当然后续操作命令也要相应地修改，最终仓库地址也有所不同（仓库地址以此用户名开头）。

网上有方法禁止git用户登录SSH。这一步是可选的。

**第3步：创建 git 仓库**

使用 git 用户登录Linux系统，在用户主目录 /home/git 内执行如下命令创建git仓库：

|  |
| --- |
| git init --bare myrepo.git |

裸仓库(–bare)目录名称以后缀 .git 结尾是一种习惯性约定，并非必需。

该仓库本地目录是 /home/git/myrepo.git，或者 ~/myrepo.git，这个毫无疑问。

该仓库 远程目录 可以是以下任意一个：

git@192.168.0.2:/home/git/myrepo.git

git@192.168.0.2:~/myrepo.git

git@192.168.0.2:myrepo.git

咱们通常使用最后一个，因为它最简洁。

注意其中包含了用户名、IP地址、本地目录等信息；注意符号 @ 和 : 都是必不可少的。

**第4步：客户端克隆 git 仓库**

Git客户端可通过如下命令克隆仓库：

|  |
| --- |
| git clone git@192.168.0.2:myrepo.git |

用户名git已经包含在仓库远程目录内，用户密码就是在第2步创建git用户时设定的那个密码。克隆仓库需要输入密码。

**第五步：客户端工程push到git仓库**

首次将本地工程push到服务器仓库时，需要把本地项目文件夹变成git仓库。

在工程路径下打开命令行，输入

|  |
| --- |
| git init |

该目录下就会生成一个.git文件夹

**第六步：提交和推送文件到远程仓库**

1. 添加所有的文件暂存

|  |
| --- |
| git add . |

1. 将暂存区的内容提交到仓库

|  |
| --- |
| git commit -m "Message" |

1. 绑定远程仓库

|  |
| --- |
| git remote add origin "git@192.168.86.66:myrepo.git" |

origin - 本地名称，可以改

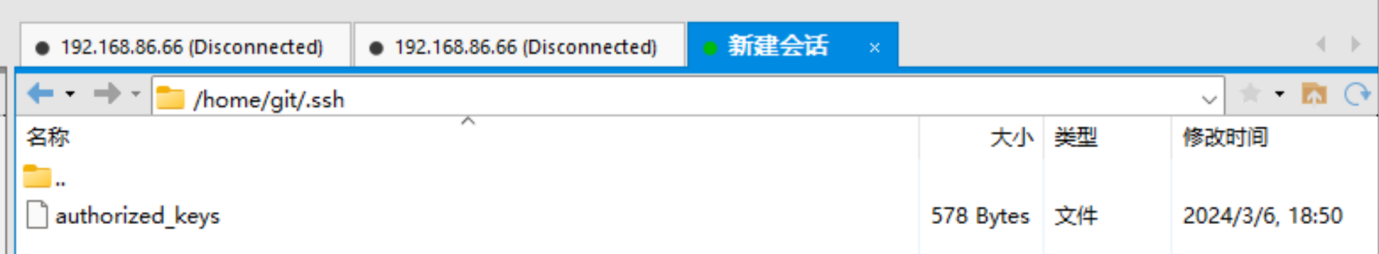
“git@192.168.86.66:myrepo.git” - “仓库链接”

1. 将本地文件推送到远程仓库

|  |
| --- |
| git push -u origin master // 推送本地文件至远程仓库 git push -u origin master -f // 强制推送本地文件至远程仓库 |

## 公钥配置

把SSH Key发给管理员，填到Git服务器的配置中，如下图。



## 问题解决：

**（1）提示“fatal: not a git repository”：**

在命令行 输入 git init 然后回车就好了

# 三、git用户权限设置

## 1. 创建用户组

首先，可以创建一个用于管理仓库权限的用户组。例如，可以使用以下命令创建一个名为”gitusers”的用户组：

|  |
| --- |
| $ sudo groupadd gitusers |

## 2. 添加用户到用户组

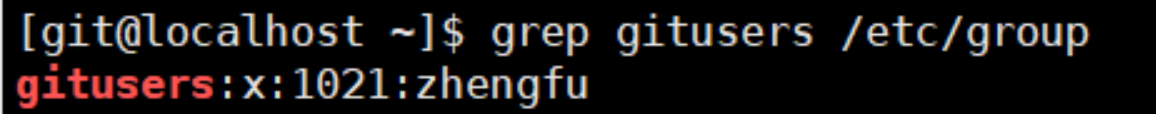
将需要管理仓库权限的用户添加到用户组中。例如，假设有一个名为”tom”的用户需要拥有仓库的读写权限，可以使用以下命令添加用户到用户组中：

|  |
| --- |
| $ sudo usermod -a -G gitusers zhengfu |

检查用户组和用户是否成功添加

|  |
| --- |
| $ grep gitusers /etc/group |

若添加成功，如下图：



修改已存在的用户组

|  |
| --- |
| # 修改已存在的gitusers用户组名称为gitusers\_new  $ groupmod -n gitusers\_new gitusers |

## 3. 创建Git仓库

使用Git命令创建一个新的仓库，或者使用现有的仓库。例如，可以使用以下命令在/home/git目录下创建一个名为”myrepo.git”的仓库：

|  |
| --- |
| $ sudo git init -bare /home/git/myrepo.git |

## 4. 设置仓库权限

使用Linux权限控制设置仓库的权限，确保只有用户组成员能够访问仓库。例如，可以使用以下命令将仓库的所有者设置为”gitusers”用户组，并将权限设置为读写（rwx）：

|  |
| --- |
| $ sudo chown -R :gitusers /home/git/myrepo.git $ sudo chmod -R g+rwX /home/git/myrepo.git |

## 5. 配置SSH密钥认证

让用户只能使用SSH密钥进行身份验证，确保只有授权的用户能够访问仓库。用户需要生成一个SSH密钥对，并将公钥添加到Git服务器上的授权密钥列表中。例如，可以使用以下命令生成SSH密钥对：

|  |
| --- |
| $ ssh-keygen -t rsa -b 4096 -C "your\_email@example.com" |

然后，将公钥（通常是`~/.ssh/id\_rsa.pub`文件）的内容添加到Git服务器上的`~/.ssh/authorized\_keys`文件中。

## 6. 配置Git仓库访问权限

使用Git命令配置仓库的访问权限，确保只有授权的用户能够克隆、拉取和推送仓库。例如，可以使用以下命令在仓库的根目录下创建一个名为”gitolite.conf”的配置文件：

|  |
| --- |
| $ sudo nano /home/git/myrepo.git/gitolite.conf |

然后，将以下内容添加到配置文件中，指定具有读写权限的用户组和仓库名称：

|  |
| --- |
| @gitusers = zhengfu repo myrepo.git  RW+ = @gitusers |

保存并退出配置文件。

## 7. 提交并推送修改

使用Git命令提交并推送刚刚对仓库进行的权限设置。例如，可以使用以下命令提交并推送修改：

|  |
| --- |
| $ cd /home/git/myrepo.git $ sudo git add . $ sudo git commit -m "Set repository permissions" $ sudo git push origin master |

通过以上步骤，就成功设置了Git仓库的权限。现在只有用户组成员才能访问、克隆、拉取和推送仓库。