



上海联天科技有限公司
Shanghai Liantian Tech Co., Ltd

Gerrit 环境初始化手册

文档标识	
当前版本	V1.0
发布状态	Release
拟 制	王 晨
批 准	魏兴朋
日 期	2021/10/28

一、Gerrit 环境初始化

新入职开发人员需要先向 SCM 申请 gerrit 账号，具体以 SCM 邮件通知为准。

1、设置 git 用户名和邮箱及 SSH Public Keys

首次登录（gerrit）需设置中文 Full Name 并绑定/验证 Email 邮箱。

The screenshot shows the Gerrit User Settings interface. On the left, there's a sidebar with links: Profile (highlighted with a red box), Preferences, Diff Preferences, Edit Preferences, Menu, Change Table Columns, Notifications, Email Addresses (highlighted with a red box), HTTP Credentials, SSH Keys, Groups, Identities, and Mail Filters. The main content area has two tabs: User Settings and Preferences. Under User Settings, there's a 'Dark theme (alpha)' checkbox, a note about it being in early alpha, and a 'Profile' section. In the Profile section, fields include ID (1000098), Email (auto-link.com.cn), Registered (Jul 29), Username (redacted), Full name (redacted), and Status (e.g. "Vacation"). A 'SAVE CHANGES' button is at the bottom. Under Preferences, settings include Changes per page (25 rows per page), Date/time format (Jun 3; Jun 3, 2016, 4:10 PM), Email notifications (Only comments left by others), Email format (HTML and plaintext), and Default Base For Merges (First Parent). A 'GENERATE NEW PASSWORD' button is also present.

本地 git 环境中

\$ ssh-keygen #会有几次提示，不用输入任何内容，全部直接回车。

\$ cat ~/.ssh/id_rsa.pub

将得到的公钥添加到 gerrit 账户的个人设置中，具体操作步骤如下：

登录 gerrit 网页客户端 <https://gerrit.auto-link.com.cn:8085>

点击右上角 Settings→SSH Public Keys，将刚生成的公钥复制粘贴并点击 Add 即可，

The screenshot shows the Gerrit SSH Keys management interface. On the left, there's a sidebar with links: Profile, Preferences, Diff Preferences, Edit Preferences, Menu, Change Table Columns, Notifications, Email Addresses, HTTP Credentials, SSH Keys (highlighted with a red box), Groups, Identities, and Mail Filters. The main content area shows a table of SSH keys. The columns are: Comment, Status, Public key, Username (wanglechen), and three action buttons: CLICK TO VIEW, DELETE, and ETC. There are several rows of keys listed, each with a different comment like 'MS-PMPNEYDRWZHK', '74K', 'REVIEWER', 'iu', and 'j'. At the bottom, there are buttons for UNLINK, CLICK TO VIEW, and ETC.

\$ echo "user **your name**">~/.ssh/config #本人用户名 （注：config 文件不存在，需要新建一下）

\$ chmod 600 ~/.ssh/config

\$ git config --global user.name **your name** #本人

\$ git config --global user.email **xxx@xx** #邮箱为本人邮箱

\$ git config --list #查看设置是否正确

\$ git config --unset --global user.email #若错误，清除后重新设置



添加`~/.ssh/config`，修改文件权限：`chmod 600 ~/.ssh/config` 文件内容如下：

```
host gerrit.auto-link.com.cn
#将 user_name 修改为自己 gerrit 账号
user user_name
hostname gerrit.auto-link.com.cn
port 29418
#需要确认使用 ssh key 路径和文件名是不是~/.ssh/id_rsa,一般默认生成不需要修改
identityfile ~/.ssh/id_rsa
```

●Linux/Windows 配置 config 使用 ssh 连接

(1) Linux

查看本地是否有 ssh 公私钥（有的话继续下一步，没有 `ssh-keygen` 生成）

```
1 cd ~/.ssh
2 ls -a
```

将公钥内容复制到要连接的服务器用户下

►方法一

```
ssh-copy-id -i /root/.ssh/id_rsa.pub root@192.168.100.10
#           |           |           |           |
#   ssh命令      本机公钥的地址  服务器用户  服务器IP地址
```

►方法二

登录到服务器，直接将本机的公钥复制到 root 下 `authorized_keys` (这里用的是 root 用户)

给`.ssh`文件夹设置权限

```
chmod 700 .ssh
```

给`authorized_keys`文件设置权限

```
chmod 600 authorized_keys
```

在 Linux 下写配置文件 config

```
touch config # 创建config文件
vi config # 编写config
```

config 具体内容

```
host "roc"          # ssh连接的别名
HostName 123.56.240.37 # 服务器IP
User root          # 登陆的用户
Port 22
```



ssh 连接

```
ssh root
```

(2) Windows

► Git Bash 配置

打开 Git Bash 查看电脑上是否已存在 SSH 密钥
输入 cd ~/.ssh

```
$ cd ~/.ssh  
bash: cd: ~/.ssh: No such file or directory
```

若显示无法找到该文件则要创建新的 ssh key:

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

```
xuyawen@CL-LAP-RD-070 MINGW64 ~/Desktop  
$ ssh-keygen -t rsa -C "xuyawen@auto-link.com.cn"  
Generating public/private rsa key pair.  
Enter file in which to save the key (/c/Users/xuyawen/.ssh/id_rsa):
```

全部回车

然后会在 .ssh 目录生产两个文件: id_rsa 和 id_rsa.pub

用记事本打开 .ssh 目录下的 id_rsa.pub 文件, 复制里面的内容;

(3) 复制 ssh key 到 GitHub

进入个人主页, 点击 settings, 进入设置页面, 选择 SSH keys 之后点击 new SSH key

测试 ssh 链接 github: 输入 ssh -T git@github.com

出现 successfully, 连接成功!

► 配置 config

跟 Linux 一样在 C 盘用户目录下有个 .ssh 文件夹

在 .ssh 文件中创建一个 config 文件

同(1)的操作把生成的公钥复制到服务器需要登录用户的 authorized_key 文件下

连接服务器



```
# 第一种
ssh rot
# 第二种
ssh root@rot # rot--> 对应Host
|
# 需要登陆的用户对应User
# 俩种方法都可以
```

☞免密登陆

在 config 文件中加入下面两横，就可以实现免密登录（同样如果 Linux 也想免密登陆的话也要加入这两横）

```
PreferredAuthentications publickey
IdentityFile C:\Users\Jack-Roc\.ssh\jack_rsa # 私钥的具体地址
```

☞解决问题

问题一

```
root@123.57.240.37: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
```

出现这个问题，使用 ssh 私钥链接

```
ssh root@rot -i ./testrsa(私钥地址) # 失败的话，检查一下你的公私钥是否一致 服务器公钥是否正确
# 检查一下文件权限
```

问题二

```
ssh: Could not resolve hostname rot: \262\273\326\252\265\300\325\342\321\371\265\304\326\367\273\372\241\243
```

说明本地 config 文件有错误

2、代码下载

代码下拉命令



2.1 用 repo 下拉代码

```
repo init -u ssh://username@20.30.90.8:29418/cluster/SCC/manifest -b 分支名 -m  
manifest.xml --repo-url=ssh://username@20.30.90.8:29418/android_gwm_v3.5/repo  
--no-repo-verify  
repo sync -c -q -j4  
repo start dev --all
```

2.1.2 本地提交

- 1.Modify your code #修改代码前请 sync 全部代码，以免出现一些不必要的问题
- 2.git add .#注意命令后面加点：“.”， add 后建议 git status 确认暂存区再提交
- 3.git commit -m " Fixed some issues or add a new feature. "

2.1.3 push 到 gerrit 代码评审系统

上传代码时无需 git push，只需执行：

```
repo upload  
# 如果只有 1 条修改，输入 y 即可自动 push 到 gerrit  
# 如果有多条修改，会调出 vim 编辑器，将需要的 push 的代码（project/branch/sha1）  
注释去掉即可自动 push  
push 后需要登录 gerrit，Add “Group Leader/SPM”来审核并 merged/review+2。
```

2.2 用 git 下拉代码

```
git clone "ssh://username@gerrit.auto-link.com.cn:29418/qnx_gac/3rdparty/svm" -b 分支名&&  
scp -p -P 29418 username@gerrit.auto-link.com.cn:hooks/commit-msg "svm/.git/hooks/"
```

2.2.2 本地提交

- 1.Modify your code #修改代码
- 2.git add .#注意命令后面加点：“.”， add 后建议 git status 确认暂存区再提交
- 3.git commit -m " Fixed some issues or add a new feature. "

2.2.3 push 到 gerrit 代码评审系统

上传代码命令：

```
git push origin HEAD:refs/for/分支名  
push 后需要登录 gerrit，Add “Group Leader/SPM”来审核并 merged/review+2。
```