

Git 全操作指南Github_SSH_Key补充 (for linux or windos system)

作者：Mr-Auqin 联系方式：2480015679@qq.com

Git 全操作指南Github_SSH_Key补充 (for linux or windos system)

文档说明

SSH 密钥配置 (ED25519 算法，免密推送 / 拉取代码，必做！)

(一) Linux 端 SSH 密钥配置 (ED25519，核心，适配远程开发)

- 步骤 1：检查是否已有 SSH 密钥（避免重复生成）
- 步骤 2：生成 ED25519 格式 SSH 密钥（核心修改）
- 步骤 3：查看并复制 SSH 公钥内容
- 步骤 4：将公钥添加到远程 Git 平台（以 GitHub 为例）
- 步骤 5：验证 SSH 连接是否成功
- 步骤 6：配置 SSH 连接持久化（可选，避免频繁重连）
- 步骤 7：Linux 端 SSH 常见问题排查

(二) Windows 端 SSH 密钥配置 (ED25519，可选，本地操作代码用)

- 步骤 1：打开 Git Bash 终端
- 步骤 2：检查是否已有 SSH 密钥
- 步骤 3：生成 ED25519 格式 SSH 密钥（核心修改）
- 步骤 4：查看并复制 SSH 公钥内容
- 步骤 5：添加公钥到远程 Git 平台
- 步骤 6：验证 SSH 连接是否成功（Git Bash 执行）
- 步骤 7：Windows 端 SSH 常见问题排查

五、通用问题排查

SSH 密钥管理 (ED25519 算法，分系统)

文档说明

作为Git 全操作指南 (VSCode 远程 Linux 版) 的Github_SSH_key 补充说明

SSH 密钥配置 (ED25519 算法，免密推送 / 拉取代码，必做！)

Git 连接远程仓库 (GitHub/Gitee/GitLab) 有两种方式：HTTPS（每次需输密码）和 SSH（免密），**推荐使用 ED25519 算法生成 SSH 密钥**（比 RSA 更安全、高效，兼容所有主流 Git 平台）。

核心说明：VSCode 远程连接 Linux 开发时，优先配置 **Linux 端 SSH 密钥**（操作 Linux 内代码时使用）；如需在 Windows 本地操作代码，需单独配置 Windows 端 SSH 密钥。

(一) Linux 端 SSH 密钥配置 (ED25519，核心，适配远程开发)

步骤 1：检查是否已有 SSH 密钥（避免重复生成）

bash

运行

```
# 进入SSH目录，查看是否有id_ed25519（私钥）、id_ed25519.pub（公钥）文件  
cd ~/.ssh  
ls -l # 若显示id_ed25519和id_ed25519.pub，说明已有ED25519密钥，可跳过生成步骤
```

步骤 2：生成 ED25519 格式 SSH 密钥（核心修改）

bash

运行

```
# 生成ED25519格式密钥（目前最安全的SSH密钥算法），-C绑定你的Git邮箱（与用户信息配置一致）  
ssh-keygen -t ed25519 -C "你的Git邮箱地址"  
# 执行后会提示：  
# Enter file in which to save the key (/home/你的用户名/.ssh/id_ed25519)：直接回车  
（使用默认路径）  
# Enter passphrase (empty for no passphrase)：可选（输入密码则每次用SSH需验证，直接回车  
=免密）  
# Enter same passphrase again：同上，直接回车
```

生成成功后，`~/.ssh` 目录会新增两个文件：

- `id_ed25519`：ED25519 私钥（核心，保存在本地，绝对不能泄露！）
- `id_ed25519.pub`：ED25519 公钥（可公开，需添加到远程 Git 平台）

补充：若你的 Linux 系统版本过旧（如 CentOS 6）不支持 ED25519，可 fallback 到 RSA 4096 位：`ssh-keygen -t rsa -b 4096 -C "你的Git邮箱地址"`

步骤 3：查看并复制 SSH 公钥内容

bash

运行

```
# 查看ED25519公钥内容（复制输出的全部文本）  
cat ~/.ssh/id_ed25519.pub  
# 示例输出格式：ssh-ed25519  
AAAAAC3NzaC1ZDI1NTE5AAAAIOMqqnkVzrm0SdG6UoqKLsabgH5C9okwi0dh219GKJ1 你的Git邮箱地  
址
```

操作：选中终端里公钥的**全部内容**（从`ssh-ed25519`开始到邮箱结束），复制（Linux 终端可右键→复制）。

步骤 4：将公钥添加到远程 Git 平台（以 GitHub 为例）

1. 登录 GitHub 账号 → 点击右上角头像 → 「Settings」
2. 左侧菜单选择「SSH and GPG keys」→ 点击「New SSH key」
3. 填写信息：
 - Title：自定义名称（如「Linux 虚拟机 - IMX6ULL 开发 - ED25519」）
 - Key type：默认「Authentication key」
 - Key：粘贴刚才复制的 ED25519 公钥内容（完整粘贴，不要漏字符）

4. 点击「Add SSH key」→ 输入 GitHub 密码验证，完成添加。

Gitee/GitLab 操作差异：

- Gitee：设置 → 安全设置 → SSH 公钥 → 添加公钥
- GitLab：设置 → SSH Keys → 添加公钥

步骤 5：验证 SSH 连接是否成功

bash

运行

```
# 验证GitHub连接（替换为对应平台域名）
ssh -T git@github.com
# 验证Gitee连接: ssh -T git@gitee.com
# 验证GitLab连接: ssh -T git@gitlab.com
```

成功提示：Hi 你的GitHub用户名！ You've successfully authenticated, but GitHub does not provide shell access.

步骤 6：配置 SSH 连接持久化（可选，避免频繁重连）

bash

运行

```
# 编辑SSH配置文件
vim ~/.ssh/config
# 粘贴以下内容（适配GitHub，其他平台可新增Host，注意IdentityFile改为ED25519私钥路径）
Host github.com
  HostName github.com
  User git
  IdentityFile ~/.ssh/id_ed25519 # 核心：指向ED25519私钥
  ServerAliveInterval 30
  ServerAliveCountMax 10
# 保存退出（Vim: Esc → :wq）
```

步骤 7：Linux 端 SSH 常见问题排查

1. 验证连接时报错「Permission denied (publickey)」：

- 检查公钥是否完整粘贴到平台，无多余空格 / 换行；
- 检查私钥权限（必须为 600）：`chmod 600 ~/.ssh/id_ed25519`；
- 重启 SSH 代理：`eval "$(ssh-agent -s)" && ssh-add ~/.ssh/id_ed25519`。

2. 公钥添加后仍需输密码：

- 确认远程仓库地址是 SSH 格式（而非 HTTPS）：

```
git remote -v
```

查看，若为 HTTPS 可修改：

bash

运行

```
git remote set-url origin git@github.com:你的用户名/仓库名.git
```

(二) Windows 端 SSH 密钥配置 (ED25519, 可选, 本地操作代码用)

步骤 1: 打开 Git Bash 终端

安装 Git 后, 在 Windows 桌面 / 文件夹空白处右键 → 选择 「Git Bash Here」。

步骤 2: 检查是否已有 SSH 密钥

bash

运行

```
# 进入SSH目录 (Windows默认路径)
cd ~/.ssh
# 或 cd C:\Users\你的用户名\.ssh
ls -l # 查看是否有id_ed25519和id_ed25519.pub文件
```

步骤 3: 生成 ED25519 格式 SSH 密钥 (核心修改)

bash

运行

```
# 生成ED25519格式密钥, -C绑定你的Git邮箱
ssh-keygen -t ed25519 -C "你的Git邮箱地址"
# 执行后提示:
# Enter file in which to save the key (/c/users/你的用户名/.ssh/id_ed25519): 直接回车
# Enter passphrase (empty for no passphrase): 直接回车(免密)
# Enter same passphrase again: 直接回车
```

步骤 4: 查看并复制 SSH 公钥内容

bash

运行

```
# 查看ED25519公钥内容
cat ~/.ssh/id_ed25519.pub
```

操作: 选中全部内容 (从 ssh-ed25519 到邮箱), 按 **ctrl+c** 复制。

备选方式: 直接打开文件复制 → 进入 `C:\Users\你的用户名\.ssh` 目录, 用记事本打开 `id_ed25519.pub`, 复制全部内容。

步骤 5: 添加公钥到远程 Git 平台

和 Linux 端步骤一致 (GitHub/Gitee/GitLab 添加公钥)。

步骤 6: 验证 SSH 连接是否成功 (Git Bash 执行)

bash

运行

```
# 验证GitHub连接  
ssh -T git@github.com  
# 验证Gitee连接  
ssh -T git@gitee.com
```

步骤 7: Windows 端 SSH 常见问题排查

1. Git Bash 执行

```
ssh
```

命令提示「command not found」：

- 确认 Git 安装路径已加入 Windows 环境变量 → 检查 `C:\Program Files\Git\usr\bin` 是否在 PATH 中。

2. 验证连接时报错「Connection timed out」：

- 检查网络代理 / 防火墙，关闭后重试；
- 切换网络（如从公司内网切到热点）。

3. 私钥权限过高 / 过低：

- Windows 下右键 `id_ed25519` 文件 → 属性 → 安全 → 高级 → 禁用继承 → 删除所有非当前用户的权限 → 仅保留当前用户的「完全控制」。

五、通用问题排查

1. Git 安装失败：

- Linux 端：确保 sudo 权限，执行 `sudo apt update` 后重试；
- Windows 端：以管理员身份运行安装包，关闭杀毒软件。

2. 用户信息配置不生效：

- 检查命令拼写（无空格 / 引号完整），重启终端后重试；
- 确认使用 `-global` 参数（全局生效）。

3. 提交时报错「Please tell me who you are」：

- 未配置用户信息，重新执行全局配置命令。

4. SSH 验证失败：

- 公钥粘贴不完整（最常见）：重新复制公钥，确保无截断 / 多余空格；
- 远程仓库地址错误：确认是 SSH 格式（`git@xxx.com:用户名/仓库名.git`）；
- ED25519 不兼容：极少数老旧平台不支持 ED25519，可改用 RSA 4096 位：`ssh-keygen -t rsa -b 4096 -C "你的邮箱"`。

##

SSH 密钥管理 (ED25519 算法, 分系统)

| 功能需求 | Linux 端终端命令 | Windows 端终端命令 (Git Bash) | 备注 |
|---------------------|--|--|--|
| 检查已有 SSH 密钥 | ls -l ~/.ssh | ls -l ~/.ssh 或 ls -l C:\Users\用户名\.ssh | 查看是否有 id_ed25519/id_ed25519.pub 文件 |
| 生成 ED25519 SSH 密钥 | ssh-keygen -t ed25519 -C "你的邮箱" | ssh-keygen -t ed25519 -C "你的邮箱" | 首选算法, 安全高效 |
| (兼容) 生成 RSA 4096 密钥 | ssh-keygen -t rsa -b 4096 -C "你的邮箱" | ssh-keygen -t rsa -b 4096 -C "你的邮箱" | 适配老旧系统 / 平台 |
| 查看 SSH 公钥内容 | cat ~/.ssh/id_ed25519.pub | cat ~/.ssh/id_ed25519.pub | 复制全部内容到远程 Git 平台 |
| 验证 GitHub SSH 连接 | ssh -T git@github.com | ssh -T git@github.com | 成功提示 "Hi 用户名! You've successfully authenticated" |
| 验证 Gitee SSH 连接 | ssh -T git@gitee.com | ssh -T git@gitee.com | 同上 |
| 配置 SSH 持久化连接 | vim ~/.ssh/config | vim ~/.ssh/config 或 notepad ~/.ssh/config | IdentityFile 指向 id_ed25519 |
| 修复私钥权限错误 | chmod 600 ~/.ssh/id_ed25519 | 右键文件→属性→安全→仅保留当前用户权限 | Linux 需 600 权限, Windows 需精简权限 |
| 加载 SSH 私钥到代理 | eval "\$(ssh-agent -s)" && ssh-add ~/.ssh/id_ed25519 | eval "\$(ssh-agent -s)" && ssh-add ~/.ssh/id_ed25519 | 解决 "Permission denied (publickey)" |
| 修改远程地址为 SSH 格式 | git remote set-url origin git@github.com : 用户名 / 仓库名.git | git remote set-url origin git@github.com : 用户名 / 仓库名.git | 适配免密操作 |

##