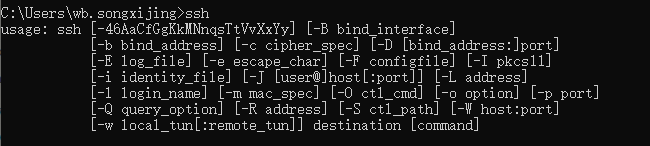
# 4.3 服务器上的 Git - 生成 SSH 公钥

## 生成 SSH 公钥



如前所述，许多 Git 服务器都使用 SSH 公钥进行认证。 为了向 Git 服务器提供 SSH 公钥，如果某系统用户尚未拥有密钥，必须事先为其生成一份。 这个过程在所有操作系统上都是相似的。 首先，你需要确认自己是否已经拥有密钥。 默认情况下，用户的 SSH 密钥存储在其 ~/.ssh 目录下。 进入该目录并列出其中内容，你便可以快速确认自己是否已拥有密钥：

$ cd ~/.ssh

$ ls

authorized\_keys2 id\_dsa known\_hosts

config id\_dsa.pub

我们需要寻找一对以 id\_dsa 或 id\_rsa 命名的文件，其中一个带有 .pub 扩展名。 .pub 文件是你的公钥，另一个则是与之对应的私钥。 如果找不到这样的文件（或者根本没有 .ssh 目录），你可以通过运行 ssh-keygen 程序来创建它们。 在 Linux/macOS 系统中，ssh-keygen 随 SSH 软件包提供；在 Windows 上，该程序包含于 MSysGit 软件包中。

$ ssh-keygen -o

Generating public/private rsa key pair.

Enter file in which to save the key (/home/schacon/.ssh/id\_rsa):

Created directory '/home/schacon/.ssh'.

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/schacon/.ssh/id\_rsa.

Your public key has been saved in /home/schacon/.ssh/id\_rsa.pub.

The key fingerprint is:

d0:82:24:8e:d7:f1:bb:9b:33:53:96:93:49:da:9b:e3 schacon@mylaptop.local

首先 ssh-keygen 会确认密钥的存储位置（默认是 .ssh/id\_rsa），然后它会要求你输入两次密钥口令。 如果你不想在使用密钥时输入口令，将其留空即可。 然而，如果你使用了密码，那么请确保添加了 -o 选项，它会以比默认格式更能抗暴力破解的格式保存私钥。 你也可以用 ssh-agent 工具来避免每次都要输入密码。

现在，进行了上述操作的用户需要将各自的公钥发送给任意一个 Git 服务器管理员 （假设服务器正在使用基于公钥的 SSH 验证设置）。 他们所要做的就是复制各自的 .pub 文件内容，并将其通过邮件发送。 公钥看起来是这样的：

$ cat ~/.ssh/id\_rsa.pub

ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAklOUpkDHrfHY17SbrmTIpNLTGK9Tjom/BWDSUGPl+nafzlHDTYW7hdI4yZ5ew18JH4JW9jbhUFrviQzM7xlELEVf4h9lFX5QVkbPppSwg0cda3Pbv7kOdJ/MTyBlWXFCR+HAo3FXRitBqxiX1nKhXpHAZsMciLq8V6RjsNAQwdsdMFvSlVK/7XAt3FaoJoAsncM1Q9x5+3V0Ww68/eIFmb1zuUFljQJKprrX88XypNDvjYNby6vw/Pb0rwert/EnmZ+AW4OZPnTPI89ZPmVMLuayrD2cE86Z/il8b+gw3r3+1nKatmIkjn2so1d01QraTlMqVSsbxNrRFi9wrf+M7Q== schacon@mylaptop.local

关于在多种操作系统中生成 SSH 密钥的更深入教程，请参阅 GitHub 的 SSH 密钥指南 <https://help.github.com/articles/generating-ssh-keys>。