**25个必须记住的SSH命令，你用过了吗？**

SSH是一个非常伟大的工具，如果你要在互联网上远程连接到服务器，那么SSH无疑是最佳的候选。SSH是加密的，OpenSSH加密所有通信（包括密码），有效消除了窃听，连接劫持和其它攻击。本文将为大家介绍25个最佳的SSH命令，希望您在阅读之后能获得一些启发。

OpenSSH是SSH连接工具的免费版本。telnet，rlogin和ftp用户可能还没意识到他们在互联网上传输的密码是未加密的，但SSH是加密的，OpenSSH加密所有通信（包括密码），有效消除了窃听，连接劫持和其它攻击。此外，OpenSSH提供了安全隧道功能和多种身份验证方法，支持SSH协议的所有版本。

SSH是一个非常伟大的工具，如果你要在互联网上远程连接到服务器，那么SSH无疑是最佳的候选。下面是通过网络投票选出的25个最佳SSH命令，你必须牢记于心。

（注：有些内容较长的命令，在本文中会显示为截断的状态。如果你需要阅读完整的命令，可以把整行复制到您的记事本当中阅读。）

**1、复制SSH密钥到目标主机，开启无密码SSH登录**

ssh-copy-id user@host

如果还没有密钥，请使用ssh-keygen命令生成。

**2、从某主机的80端口开启到本地主机2001端口的隧道**

ssh -N -L2001:localhost:80 somemachine

现在你可以直接在浏览器中输入http://localhost:2001访问这个网站。

**3、将你的麦克风输出到远程计算机的扬声器**

dd if=/dev/dsp | ssh -c arcfour -C username@host dd of=/dev/dsp

这样来自你麦克风端口的声音将在SSH目标计算机的扬声器端口输出，但遗憾的是，声音质量很差，你会听到很多嘶嘶声。

**4、比较远程和本地文件**

ssh user@host cat /path/to/remotefile | diff /path/to/localfile –

在比较本地文件和远程文件是否有差异时这个命令很管用。

**5、通过SSH挂载目录/文件系统**

sshfs name@server:/path/to/folder /path/to/mount/point

从<http://fuse.sourceforge.net/sshfs.html>下载sshfs，它允许你跨网络安全挂载一个目录。

**6、通过中间主机建立SSH连接**

ssh -t reachable\_host ssh unreachable\_host

Unreachable\_host表示从本地网络无法直接访问的主机，但可以从reachable\_host所在网络访问，这个命令通过到reachable\_host的“隐藏”连接，创建起到unreachable\_host的连接。

**7、将你的SSH公钥复制到远程主机，开启无密码登录 – 简单的方法**

ssh-copy-id username@hostname

**8、直接连接到只能通过主机B连接的主机A**

ssh -t hostA ssh hostB

当然，你要能访问主机A才行。

**9、创建到目标主机的持久化连接**

ssh -MNf <user>@<host>

在后台创建到目标主机的持久化连接，将这个命令和你~/.ssh/config中的配置结合使用：

Host host

ControlPath ~/.ssh/master-%r@%h:%p

ControlMaster no

所有到目标主机的SSH连接都将使用持久化SSH套接字，如果你使用SSH定期同步文件（使用rsync/sftp/cvs/svn），这个命令将非常有用，因为每次打开一个SSH连接时不会创建新的套接字。

**10、通过SSH连接屏幕**

ssh -t remote\_host screen –r

直接连接到远程屏幕会话（节省了无用的父bash进程）。

**11、端口检测（敲门）**

knock <host> 3000 4000 5000 && ssh -p <port> user@host && knock <host> 5000 4000 3000

在一个端口上敲一下打开某个服务的端口（如SSH），再敲一下关闭该端口，需要先安装knockd，下面是一个配置文件示例。

[options]

logfile = /var/log/knockd.log

[openSSH]

sequence = 3000,4000,5000

seq\_timeout = 5

command = /sbin/iptables -A INPUT -i eth0 -s %IP% -p tcp –dport 22 -j ACCEPT

tcpflags = syn

[closeSSH]

sequence = 5000,4000,3000

seq\_timeout = 5

command = /sbin/iptables -D INPUT -i eth0 -s %IP% -p tcp –dport 22 -j ACCEPT

tcpflags = syn

**12、删除文本文件中的一行内容，有用的修复**

ssh-keygen -R <the\_offending\_host>

在这种情况下，最好使用专业的工具。

**13、通过SSH运行复杂的远程shell命令**

ssh host -l user $(<cmd.txt)

更具移植性的版本：

ssh host -l user “`cat cmd.txt`”

**14、通过SSH将MySQL数据库复制到新服务器**

mysqldump –add-drop-table –extended-insert –force –log-error=error.log -uUSER -pPASS OLD\_DB\_NAME | ssh -C user@newhost “mysql -uUSER -pPASS NEW\_DB\_NAME”

通过压缩的SSH隧道Dump一个MySQL数据库，将其作为输入传递给mysql命令，我认为这是迁移数据库到新服务器最快最好的方法。

**15、删除文本文件中的一行，修复“SSH主机密钥更改”的警告**

sed -i 8d ~/.ssh/known\_hosts

**16、从一台没有SSH-COPY-ID命令的主机将你的SSH公钥复制到服务器**

cat ~/.ssh/id\_rsa.pub | ssh user@machine “mkdir ~/.ssh; cat >> ~/.ssh/authorized\_keys”

如果你使用Mac OS X或其它没有ssh-copy-id命令的\*nix变种，这个命令可以将你的公钥复制到远程主机，因此你照样可以实现无密码SSH登录。

**17、实时SSH网络吞吐量测试**

yes | pv | ssh $host “cat > /dev/null”

通过SSH连接到主机，显示实时的传输速度，将所有传输数据指向/dev/null，需要先安装pv。

如果是Debian：

apt-get install pv

如果是Fedora：

yum install pv

（可能需要启用额外的软件仓库）。

**18、如果建立一个可以重新连接的远程GNU screen**

ssh -t user@some.domain.com /usr/bin/screen –xRR

人们总是喜欢在一个文本终端中打开许多shell，如果会话突然中断，或你按下了“Ctrl-a d”，远程主机上的shell不会受到丝毫影响，你可以重新连接，其它有用的screen命令有“Ctrl-a c”（打开新的shell）和“Ctrl-a a”（在shell之间来回切换），请访问http://aperiodic.net/screen/quick\_reference阅读更多关于screen命令的快速参考。

**19、继续SCP大文件**

rsync –partial –progress –rsh=ssh $file\_source $user@$host:$destination\_file

它可以恢复失败的rsync命令，当你通过VPN传输大文件，如备份的数据库时这个命令非常有用，需要在两边的主机上安装rsync。

rsync –partial –progress –rsh=ssh $file\_source $user@$host:$destination\_file local -> remote

或

rsync –partial –progress –rsh=ssh $user@$host:$remote\_file $destination\_file remote -> local

**20、通过SSH W/ WIRESHARK分析流量**

ssh root@server.com ‘tshark -f “port !22″ -w -' | wireshark -k -i –

使用tshark捕捉远程主机上的网络通信，通过SSH连接发送原始pcap数据，并在wireshark中显示，按下Ctrl+C将停止捕捉，但也会关闭wireshark窗口，可以传递一个“-c #”参数给tshark，让它只捕捉“#”指定的数据包类型，或通过命名管道重定向数据，而不是直接通过SSH传输给wireshark，我建议你过滤数据包，以节约带宽，tshark可以使用tcpdump替代：

ssh root@example.com tcpdump -w – ‘port !22′ | wireshark -k -i –

**21、保持SSH会话永久打开**

autossh -M50000 -t server.example.com ‘screen -raAd mysession’

打开一个SSH会话后，让其保持永久打开，对于使用笔记本电脑的用户，如果需要在Wi-Fi热点之间切换，可以保证切换后不会丢失连接。

**22、更稳定，更快，更强的SSH客户端**

ssh -4 -C -c blowfish-cbc

强制使用IPv4，压缩数据流，使用Blowfish加密。

**23、使用cstream控制带宽**

tar -cj /backup | cstream -t 777k | ssh host ‘tar -xj -C /backup’

使用bzip压缩文件夹，然后以777k bit/s速率向远程主机传输。Cstream还有更多的功能，请访问http://www.cons.org/cracauer/cstream.html#usage了解详情，例如：

echo w00t, i’m 733+ | cstream -b1 -t2

**24、一步将SSH公钥传输到另一台机器**

ssh-keygen; ssh-copy-id user@host; ssh user@host

这个命令组合允许你无密码SSH登录，注意，如果在本地机器的~/.ssh目录下已经有一个SSH密钥对，ssh-keygen命令生成的新密钥可能会覆盖它们，ssh-copy-id将密钥复制到远程主机，并追加到远程账号的~/.ssh/authorized\_keys文件中，使用SSH连接时，如果你没有使用密钥口令，调用ssh user@host后不久就会显示远程shell。

**25、将标准输入（stdin）复制到你的X11缓冲区**

ssh user@host cat /path/to/some/file | xclip

你是否使用scp将文件复制到工作用电脑上，以便复制其内容到电子邮件中？xclip可以帮到你，它可以将标准输入复制到X11缓冲区，你需要做的就是点击鼠标中键粘贴缓冲区中的内容。