### **1、查看SSH客户端版本**

有的时候需要确认一下SSH客户端及其相应的版本号。使用ssh -V命令可以得到版本号。需要注意的是，Linux一般自带的是OpenSSH: 下面的例子即表明该系统正在使用OpenSSH：

$ ssh -V   
OpenSSH\_3.9p1, OpenSSL 0.9.7a Feb 19 2003

下面的例子表明该系统正在使用SSH2：

$ ssh -V   
ssh: SSH Secure Shell 3.2.9.1 (non-commercial version) on i686-pc-linux-gnu

### **2、连接到远程主机：**

命令格式 ：  
ssh name@remoteserver 或者  
ssh remoteserver -l name  
说明：以上两种方式都可以远程登录到远程主机，server代表远程主机，name为登录远程主机的用户名。

### **3、连接到远程主机指定的端口：**

命令格式：  
ssh name@remoteserver -p 2222 或者  
ssh remoteserver -l name -p 2222  
说明：p 参数指定端口号，通常在路由里做端口映射时，我们不会把22端口直接映射出去，而是转换成其他端口号，这时就需要使用-p端口号命令格式。

### **4、通过远程主机1跳到远程主机2：**

命令格式：  
ssh -t remoteserver1 ssh remoteserver2  
说明：当远程主机remoteserver2无法直接到达时，可以使用-t参数，然后由remoteserver1跳转到remoteserver2。在此过程中要先输入remoteserver1的密码，然后再输入remoteserver2的密码，然后就可以操作remoteserver2了。

### **5、通过SSH运行远程shell命令：**

命令格式：  
ssh -l name remoteserver ‘command’  
说明：连接到远程主机，并执行远程主机的command命令。例如：查看远程主机的内存使用情况。  
$ ssh -l root 192.168.1.100 svmon -G

### **6、修改SSH监听端口：**

默认情况下，SSH监听连接端口22，攻击者使用端口扫描软件就可以看到主机是否运行有SSH服务，将SSH端口修改为大于1024的端口是一个明智的选择，因为大多数端口扫描软件（包括nmap）默认情况都不扫描高位端口。打开/etc/ssh/sshd\_config文件并查找下面这样的行：  
Port 22  
去掉该行前面的# 号，然后修改端口号并重新启动SSH服务：  
$ /etc/init.d/ssh restart

### **7、仅允许SSH协议版本2：**

有两个SSH协议版本，仅使用SSH协议版本2会更安全，SSH协议版本1有安全问题，包括中间人攻击（man-in-the-middle）和注入（insertion）攻击。编辑/etc/ssh/sshd\_config文件并查找下面这样的行：  
# Protocol 2，1  
修改为  
Protocol 2

### **8、禁止root用户登录：**

通常情况下，不采用直接用root用户登录到远程主机，由于root用户拥有超级权限，这样会带来安全隐患，所以，一般我们用普通用户登录，当需要管理远程主机时，再切换到root用户下。打开/etc/ssh/sshd\_config文件并查找下面这样的行：  
#PermitRootLogin yes  
将#号去掉，然后将yes修改成no，重启ssh服务，这样就可以禁止root用户登录。  
  
将#号去掉，然后将bannertest.txt文件的全路径替换/some/path，然后保存，重启ssh服务。当客户端登录时，就会看到bannertest.txt文件中的提示信息。

### **9、进行端口映射：**

假如公司内网有台web服务器，但是只对内不对外，这样，外网就无法访问，可以用ssh进行端口映射来实现外网访问内网的web服务器。假如web服务器名为webserver，webserver可以用ssh访问到远端主机remoteserver，登录到webserver，然后用下面命令进行映射  
命令格式：  
ssh -R 3000:localhost:80 remoteserver  
执行完成后，在remoteserver机器上，执行netstat -an | grep 3000，查看有没有开通3000端口。并执行以下命令观察是否可以打开webserver上的网页  
$ w3m http://127.0.0.1:3000  
如果能打开界面，说明映射成功.但是，这只限于本机访问web服务器，即只能remoteserver机器访问webserver。因为3000端口绑定的是remoteserver机器的127.0.0.1端口。可以编辑remoteserver机器上的/etc/ssh/sshd\_config文件并添加如下内容：  
添加 GatewayPorts yes 内容，把监听端口3000绑定到 0.0.0.0 地址上，这样外部的所有机器都能访问到这个监听端口，然后保存退出。并重启ssh服务。完成后其它机器就可以在浏览器中输入 http://remoteserver:3000来访问webserver了。

### **10、设置登录时提示信息**

首先编辑一个文件，如bannertest.txt，文件内容自行定义。然后打开/etc/ssh/sshd\_config文件并查找下面这样的行：  
#Banner /some/path

目录操作：

rm -rf mydir /\*删除mydir目录\*/

mkdir dirname /\*创建名为dirname的目录\*/

cd mydir /\*进入mydir目录\*/

cd – /\*回上一级目录\*/

cd .. /\*回父目录，中间有空格\*/

cd ~ /\*回根目录\*/

mv tools tool /\*把tools目录改名为tool \*/

ln -s tool bac /\*给tool目录创建名为bac的符号链接,最熟悉的应该就是FTP中www链接到public\_html目录了\*/

cp -a tool /home/vpser/www /\*把tool目录下所有文件复制到www目录下 \*/

文件操作：

rm go.tar /\* 删除go.tar文件 \*/

find mt.cgi /\* 查找文件名为mt.cgi的文件 \*/

df –h /\* 查看磁盘剩余空间,好像没这个必要，除非你太那个了 \*/

解压缩：

tar xvf wordpress.tar /\* 解压tar格式的文件 \*/

tar -tvf myfile.tar /\* 查看tar文件中包含的文件 \*/

tar cf toole.tar tool /\* 把tool目录打包为toole.tar文件 \*/

tar cfz vpser.tar.gz tool /\* 把tool目录打包且压缩为vpser.tar.gz文件，因为.tar文件几乎是没有压缩过的，MT的.tar.gz文件解压成.tar文件后差不多是10MB \*/

tar jcvf /var/bak/www.tar.bz2 /var/www/ /\*创建.tar.bz2文件，压缩率高\*/

tar xjf www.tar.bz2 /\*解压tar.bz2格式\*/

gzip -d ge.tar.gz /\* 解压.tar.gz文件为.tar文件 \*/

unzip phpbb.zip /\* 解压zip文件，windows下要压缩出一个.tar.gz格式的文件还是有点麻烦的 \*/

下载文件：

wget http://soft.vpser.net/web/nginx/nginx-0.8.0.tar.gz

/\*下载远程服务器上的文件到自己的服务器，连上传都省了，服务器不是100M就是1000M的带宽，下载一个2-3兆的MT还不是几十秒的事 \*/

wget -c http://soft.vpser.net/web/nginx/nginx-0.8.0.tar.gz

/\* 继续下载上次未下载完的文件 \*/

进程管理：

ps -aux /\*ps 进程状态查询命令\*/

ps命令输出字段的含义：

[list]

[\*]USER，进程所有者的用户名。

[\*]PID，进程号，可以唯一标识该进程。

[\*]%CPU，进程自最近一次刷新以来所占用的CPU时间和总时间的百分比。

[\*]%MEM，进程使用内存的百分比。

[\*]VSZ，进程使用的虚拟内存大小，以K为单位。

[\*]RSS，进程占用的物理内存的总数量，以K为单位。

[\*]TTY，进程相关的终端名。

[\*]STAT，进程状态，用(R–运行或准备运行；S–睡眠状态；I–空闲；Z–冻结；D–不间断睡眠；W-进程没有驻留页；T停止或跟踪。)这些字母来表示。

[\*]START，进程开始运行时间。

[\*]TIME，进程使用的总CPU时间。

[\*]COMMAND，被执行的命令行。

[/list]

ps -aux | grep nginx /\*在所有进程中，查找nginx的进程\*/

kill 1234 /\*1234为进程ID，即ps -aux 中的PID\*/

killall nginx /\*killall 通过程序的名字，直接杀死所有进程，nginx为进程名\*/

Vim操作：

移动类的：

h/j/k/l: 左/下/上/右　移一格

w : 向后词移动　（前面加数字移动多少个词）

b : 向前词移动　（前面加数字移动多少个词）

e : 向后移到词末

ge : 向前移到词末

$ <End> : 行末

<Home> : 行首

tx : 向右查找本行的x并移到那儿（大写时向左）

33G : 移到文件的第33行

gg : 文件首行

G : 文件尾行

33% : 文件的33%处

H/M/L : 屏幕的首/中/尾行

zt/zz/zb : 当前行移到屏幕的首/中/底部

跳转：

” : 回到跳转来的地方

CTRL-O : 跳到一个 “较老” 的地方

CTRL-I <Tab> : 则跳到一个 “较新” 的地方

查找：

/ : 向下查找（后加关键字）

? : 向上查找（后加关键字）

n : 下一条符合的记录

编辑：

i : 转换到插入模式

x : 删除当前字符

. : 重复最后一次的修改操作(同PS里ctrl+f执行滤镜)

u : 撤销操作

CTRL-R : 重做

p : 将删除的字符插入到当前位置(put)

退出保存：

:q : 退出

:q! : 不保存退出

ZZ : 保存后退出

:e! : 放弃修改重新编辑