1 Secure Remote Login

Here is a situation you might face. You are away from home, but sometimes want to log in to your home computer to get some data from your own 'cloud' or view data collected whilst away or for all sorts of other reasons. Assume that your home computer has been set to boot regularly and stay available for a few hours. This tutorial looks in detail at the command-line configuration necessary to connect securely, and this diagram shows the general situation from start to finish.

```
an ISP   >>>>>>>   INTERNET   >>>>>>>>>     your ISP
^                                        some ignore ports<1024
^                                        ---------------------------
^                                        your IP --> 201.202.203.204
^                                        modem/router
^                                        passes port 55555
^                                        thru to port 55555
^                                        and out to -> 192.168.1.2
^                                        modem IP --> 192.168.1.1
^                                        ---------------------------
^                                        home LAN IP-> 192.168.1.2
^                                        firewall rules
^                                        ssh daemon config.
^                                        ^
you@AWAY COMPUTER                        you@HOME COMPUTER
```

1.1 Arrange for passwordless login before doing this.

Assume you login to the same username (*say* `you`) on each computer. Connect the `away` computer and the `home` computer by ethernet cable and pair them cryptographically by running:

```
away~# apt-get install openssh-server telnet expect
away~$ ssh-keygen
away~$ ssh-copy-id $home
```

1.2 O.K., let's get started !!

```
away~$ ssh you@home
ssh: Could not resolve hostname home: Name or service not known
away~$
```

1.3 Prepare the home computer by connecting it to the internet.

Pre-configure your modem to hand out a fixed DHCP IP address (*eg* `192.168.1.2`) and run:

```
home~$ sudo /sbin/dhclient -r eth0
home~$ sudo /sbin/dhclient eth0
bound to 192.168.1.2 -- renewal in 38290 seconds.
home~$
```

1.4 Find the internet-facing IP address of the home computer.

```
home~$ sudo /usr/bin/arp-scan --interface=eth0 192.168.1.1/24 2>/dev/null
192.168.1.1 xx:xx:xx:xx:xx:xx DynaLink Modem
home~$ telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.
DynaLink ADSL2+ Wireless Router
Login name: admin
Password: [password]
 > ifconfig eth0
eth0 Link encap:Ethernet  HWaddr xx:xx:xx:xx:xx:xx
     inet addr:201.202.203.204  Bcast:201.202.203.255  Mask:255.255.255.0
 > logout
Bye bye. Have a nice day!!!
Connection closed by foreign host.
home~$
```

1.5 Really need to get the IP address without any interaction.

```
home~$ cat > execute-modem-command << eof
> #!/usr/bin/expect
> spawn telnet 192.168.1.1
> set env(TERM) vt100
> set timeout 4
> expect "Login name:"
> send "admin\r"
> expect "Password:"
> send "password\r"
> expect ">"
> send "ifconfig eth0\r"
> expect ">"
> send "logout\r"
> close
> eof
home~$ chmod +x execute-modem-command
home~$ ./execute-modem-command|grep "inet addr"|tr ':' ' '|awk '{print $3}'
201.202.203.204
home~$
away~$ ssh you@201.202.203.204         [hangs]
^C
away~$
```

1.6 You can obtain some debugging information (-v option).

```
away~$ ssh -v you@201.202.203.204
```

```
OpenSSH   ...   OpenSSL 1.0.1k 8 Jan 2015
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Connecting to 201.202.203.204 port 22.
[hangs]
^C
away~$
```

1.7 Occasional ISP blocks the old UNIX root login ports 1 to 1023.

```
away~$ ssh -v you@201.202.203.204
OpenSSH   ...   OpenSSL 1.0.1k 8 Jan 2015
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Connecting to 201.202.203.204 port 22.
ssh: connect to host 201.202.203.204 port 22: Connection refused
away~$
```

1.8 Try a port above 1023 to sidestep possible ISP restrictions.

```
away~$ ssh -v -p 55555 you@201.202.203.204
OpenSSH_  ...  OpenSSL 1.0.1k 8 Jan 2015
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Connecting to 201.202.203.204 port 55555.
ssh: connect to host 201.202.203.204 port 55555: No route to host
away~$
```

1.9 Configure modem to pass port 55555 on to home computer.

```
home~$ browser 192.168.1.1
[browser starts]
 username and password are being requested by http://192.168.1.1.
 user: admin
 password: password
[click something like "Advanced Setup"]
[click something like "NAT" and/or click something like "Virtual Servers"]
[click an "Add" button and enter as follows:]
Server External Port    Protocol Internal Port   Server IP    WAN
Name   Start     End             Start    End    Address      Interface
 ssh   55555     55555  TCP      55555    55555  192.168.1.2  ppp0
[quit browser]

away~$ ssh -v -p 55555 you@201.202.203.204
OpenSSH  ...  OpenSSL 1.0.1k 8 Jan 2015
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: Connecting to 201.202.203.204 port 55555.
debug1: connect to address 201.202.203.204 port 55555: No route to host
ssh: connect to host 201.202.203.204 port 55555: No route to host
```

1.10 Configure the ssh daemon to handle incoming ssh on port 55555.

```
home~# edit /etc/ssh/sshd_config
[add or amend these three lines as appropriate:]
ListenAddress 192.168.1.2:55555
PermitRootLogin no
PasswordAuthentication no
home~# /etc/init.d/ssh restart
Restarting OpenBSD Secure Shell server: sshd.
home~$
```

1.11 Amend your firewall rules to accept incoming ssh on port 55555.

```
home~# edit /usr/local/bin/iptables.init
[add a line something like this:]
iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 55555 -j ACCEPT
home~# /sbin/iptables -F
home~# /usr/local/bin/iptables.init
home~#
```

1.12 Now try for your first successful login !!

```
away~$ ssh -p 55555 you@201.202.203.204
The authenticity of host '201.202.203.204:55555' can't be established.
RSA key fingerprint is xx.xx.xx ... xx.xx.xx
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '201.202.203.204:55555' (RSA) to the list of known hosts.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY.
    You're in!  Congratulations! and Welcome :-)
home~$ exit
Connection to 201.202.203.204 closed.
away~$
```

2 Automatic and Convenient Home Computer Preparation.

This is accomplished by inserting a USB stick, which /etc/rc.local mounts on booting, and executes an initialisation program (`remote.init`) while still running as root, if present on the stick:

```
home~$ cat > remote.init << eof
> /sbin/dhclient -r eth0
> /sbin/dhclient eth0
> IP=$(execute-modem-command|grep pppoe|awk '{print $NF}')
> /usr/bin/mutt -s "$IP" you@your-isp < /dev/null
> /etc/init.d/ssh restart
> /usr/local/bin/iptables.init
> exit 0
> eof
```