

Re: reverse SSH / SSH over NAT traversal

Source: <http://linux.derkeiler.com/Mailing-Lists/Fedora/2007-09/msg00535.html>

- *From:* Konstantin Svist <fry.kun@xxxxxxxxxx>
 - *Date:* Wed, 05 Sep 2007 09:45:43 -0700
-

Jeffrey Ross wrote:

Konstantin Svist wrote:

Hi all,

I'm trying to connect to a remote host to do some simple support. The remote host is behind a NAT/firewall and it's not possible to ask the admin to tunnel a port. The remote host has a live person working on it (a linux newbie). The question is, can the remote user type in some command in their terminal to connect (SSH?) to my network – and thereby allow me to get a terminal on the remote machine. I think this is possible with a reverse SSH tunnel – but I don't really want to allow the remote user any access to my system. It's probably possible to set up some chroot or otherwise locked out environment (/bin/nologin ?), but I want to first check if there are any simpler options.

A better question: is it possible to establish an SSH connection if both networks have NAT/firewalls that can't be easily controlled? I know some programs (e.g. skype) are able to traverse NATs by various means (UDP). Can some tunnel of this sort be established so that an SSH connection can be established on top of that?

Thanks!

I've never tried using chroot for anything but to handle the ssh issue have the user enter the command:

```
ssh -R 12345:127.0.0.1:22 username@yourhost
```

have the user login. At this point you can enter:

```
ssh -p 12345 username@xxxxxxxxxx
```

Re: reverse SSH / SSH over NAT traversal

where username is the username on your clients machine.

If you want to take it one step further you can enter:

```
ssh -D 4567 -p 12345 username@xxxxxxxx
```

now your local machine is running as if its a socks proxy and any software that can be told to use a proxy will be able to connect to the remote network, set the proxy host to be 127.0.0.1 and the port to be 4567 (you can adjust the port if you like)

What I do for my own use is use the "-D" option and a program I found called "connect"
<http://www.taiyo.co.jp/~gotosh/ssh/connect.c>
and then put in ~/.ssh/config these two lines:

```
host 10.*  
ProxyCommand /usr/local/bin/connect -4 -S 127.0.0.1:4567 %h %p
```

you can do "host *.foo.com" as well and it will match anything in the .foo.com domain.

Now every time I type "ssh username@xxxxxxxx" it automatically proxies my connection if the ssh tunnel is up.

Jeff

Doesn't allowing the other user to create an SSH tunnel lower your security? They might append a -L option (when they do ssh -R) and - presto - they have unfirewalled access to your ports. Granted, this is usually not an issue when users on the other side are newbies - but if you get used to this technique and use it when it's not safe... you get the point

--

fedora-list mailing list

fedora-list@xxxxxxxx

To unsubscribe: <https://www.redhat.com/mailman/listinfo/fedora-list>