

sftp through other ssh server (proxy like)

Asked 9 years, 11 months ago Modified 9 years, 7 months ago Viewed 4k times



1



When I access a server from my local network, I first have to connect to my main server (all ports go to this machine) and when I'm on my main server, I have to connect to my next machine (which is only available from my local network).

Example:

```
name@mylaptop:~$ ssh name@example.com
[Entering Password]
Logged in
name@mainserver:~$ ssh 192.168.0.1
[Entering password]
Logged in
name@otherserver:~$
```

What I want to do now is connecting with FileZilla to the main server and "proxiyng" to the "otherserver". How can I tell FileZilla to use a command at the beginning? Or is this even possible?

ssh

proxy

sftp

Share Improve this question Follow

edited Sep 9, 2013 at 3:29



slm

9,959 ● 10 ● 49 ● 57

asked Sep 8, 2013 at 18:52



feedc0de

113 ● 1 ● 6

2 Answers

Sorted by: Highest score (default) 



I know this question was asked a few months ago - but when searching for answers to a similar requirement I had I found use of OpenSSH's ProxyCommand very helpful.

3

I have something like the following setup in my local client config file (I in fact needed three hops to get to my target: external server-> router -> target box on private network)

```
In file ~/.ssh/config

Host myexternalserver
  User username1
  IdentityFile <key file for username1@myexternalserver>
  Hostname myexternalserver.fully.qualified.name

Host myrouter
  Hostname myrouter.fully.qualified.name
  User root
  IdentityFile <key file for root@myrouter>
  ProxyCommand ssh -q myexternalserver nc %h %p

Host mytarget
  Hostname mytarget.private.name
  User username2
  IdentityFile <key file for username2@mytarget>
  ProxyCommand ssh -q myrouter nc %h %p
```

This chain means I can just type

```
ssh mytarget
```

and be sent through the required hops.

All the key files are stored locally so you don't need to put files on external servers. The same config even works for scp:

```
scp <localfile> mytarget:<remotepath>
```

A number of alternative approaches to ssh proxying are documented on the following wikibooks page:

http://en.wikibooks.org/wiki/OpenSSH/Cookbook/Proxies_and_Jump_Hosts

Share Improve this answer Follow

answered Feb 5, 2014 at 13:49



AlexMF

46 ● 2