SSH through HTTPS

The following guide describes how to set up a SSH connection with tunnels through a HTTPS connection. As long as HTTP on port 80 and HTTPS on port 443 is enabled, you can establish a SSH connection.

Setup

Other setups than the one explained below are of course possible, but here is an overview on how I set up my environment.

1. SSH connection to a linux server running Debian 7 and Ubuntu 12.04
2. SSH public key authentication, login with password is disabled
3. Tunnels to services on my server, for example the web interface of BackupPC
4. Transfer files from and to the server with WinSCP
5. You will need a google account to be able to use the http proxy

Installation

The connection will be established through your browser. At this time, I got it working with Chrome only.

1. Download Chrome and install the Secure Shell plugin.
2. Open Secure Shell in Chrome and enter the following settings:

```
free form text (line 1): name your connection, for example the server's
name
username: <user>
hostname: <www.mydomain.com>
port: <port> (note: this is the SSH port on the server)
relay options: --proxy-host=relay.wsn.at --proxy-port=443 --use-ssl
Identy: <your SSH keypair>
SSH Arguments: -L 7000:anothermachine:80 -L 22:localhost:22
Terminal Profile: leave at default or give the current profile any name
```

3. Note on SSH keypair: upload your SSH keypair, e.g. id_rsa and id_rsa.pub generated on your server (you might want to rename the files to *myserver_id_rsa* and *myserver_id_rsa.pub*, if you want to establish SSH connections to more than one machine)
4. Note on SSH Arguments: this is just an example. The first -L option establishes a tunnel to anothermachine's HTTP port through port 7000 on your client from where you initiate the connection, the second option establishes a tunnel to the server's SSH service you are connecting to. You will need this if you want to exchange files with the server through WinSCP
5. Download *puttygen.exe* from the PuTTY download page. Convert *myserver_id_rsa* to *myserver.ppk*, as WinSCP requires a PuTTY formatted private key.
6. Download WinSCP and enter the following settings:

```
File protocol: SFTP
Host name: <www.mydomain.com>
Port: 22
User name: <user>
Click button Advanced, then select your myserver.ppk file under
SSH->Authentication
```

Now, open Secure Shell in Chrome, establish the connect, then start WinSCP and connect. After a short while, you should see your remote directory.

Links

- Chrome download
- Secure Shell plugin
- WinSCP download
- PuTTY download
- nassh-relay documentation and download
- nassh-relay
- Web-based SSH
- SSH/OpenSSH/PortForwarding
- SSH Through or Over Proxy
- List of TCP and UDP port numbers

From:
https://wiki.condrau.com/ - **Bernard's Wiki**

Permanent link:
https://wiki.condrau.com/comp:httpssh

Last update: **2016/01/14 23:25**