Setting up SSH Tunneling - Windows

Setting up SSH Tunneling

In computer networking, a computer decides what to do with an incoming network packet according to the "port" it arrived on. The port is simply a number attached to the packet. Certain ports are reserved for specific functions, for example packets arriving on port 22 are assumed to be intended for the SSH handler, so the computer passes those packets to SSH to interpret. Other port numbers are available to use for whatever you like, and as long as the same port is not used for different things on the same computer, everything works.

With SSH Tunneling, you will start an SSH session between your workstation and the bastion host hpc.nyu.edu, and instruct that session to create a tunnel. Your workstation will make one end of the tunnel, at "localhost, port 8023" ("localhost" is the computer’s name for itself, so packets arriving at your workstation port 8023 will be sent into the tunnel). The bastion host will make the other end of the tunnel, at "prince.hpc.nyu.edu, port 22", so anything coming through the tunnel will be forwarded to the normal SSH port (22) of Prince. The fact that your workstation cannot see Prince does not matter, it only needs to see its end of the tunnel.

The following diagram illustrates the process. It looks complex, but only requires 2 steps: the blue text shows what happens when you create the tunnel (step 1) and the green arrows indicate using the tunnel (step 2).

You only need to do step 1 once, and then you can use the tunnel (step 2) as many times as you like - for example, you might have two terminal sessions and a WinSCP session all using the same tunnel created with step 1.

In these instructions we are using port 8023. If it happens that another program on your computer is watching this port (which is fairly unlikely) then it won’t work, and you’ll need to choose a different port number, eg 9020, and substitute that throughout these instructions. 4-digit numbers starting with an 8 or a 9 are usually good ones to choose.

Step 1: Creating the tunnel

1. First open Putty and prepare to log in to hpc.nyu.edu, as you did in Logging in from Windows - primitive way. If you saved your session
1. During that process, you can load it by selecting from the "Saved Sessions" box and hitting "Load". Don't hit "Open" yet!
2. Under "Connection" -> "SSH", just below "X11", select "Tunnels"

3. Select "Local ports accept connections from other hosts"
4. Write "8023" (the port number) in the "Source port" box, and "prince.hpc.nyu.edu:22" (the machine you wish to tunnel to - 22 is the port that ssh listens on) in the "Destination" box.
5. Click "Add". You can repeat step 4 with a different port number and a different destination, if you like (for instance, Babar). If you do this you will create multiple tunnels, one to each destination.
6. Before hitting "Open", go back to the "Sessions" page, give the session a name ("hpctunnel") and hit "Save". Then next time you need not do all this again, just load the saved session.
7. Hit "Open" to login in to hpc.nyu.edu and create the tunnel. A terminal window will appear, asking for your login name (NYU NetID) and password (NYU password). Windows may also ask you to allow certain connections through its firewall - this is so you can ssh to port 8023 on your workstation - the entrance to the tunnel.

You can add other NYU hosts to the tunnel by adding a new source port and destination and clicking "Add". For example, Prince has 4 login hosts, if you specifically need to use number 1, you could add "Source port = 8024" and "Destination = prince1.hpc.nyu.edu:22", then press "Add". You would then perform Step 2 (below) twice - once for Prince on port 8023 and once for Prince1 on port 8024.