How do I set up an SSH tunnel?

Background

There are times when you want to access a service on an IU system but either that service is not accessible remotely or the service is not sufficiently secure. You can solve both of these issues by using what is called an SSH tunnel. For example, let’s say you need to access a MySQL server that is running on the standard port 3306 but IU blocks that port at the border routers. Or, you are running some service on your local workstations but the local firewall rules are preventing you from accessing that service remotely. If you have access to the SSH port 22 on the system then you can access other ports on the remote system using SSH tunnelling.

Linux and Mac

Let’s imagine that you have mysql running on a computer named soic_machine.soic.indiana.edu. Mysql is listing on port 3306 but that port is blocked at the IU border routers so it not directly accessible from outside IU. So, we can use SSH tunnelling to reach that port. On your local Linux or Mac system, open up a terminal window and run the following:

```
ssh -4 -N -L 3306:localhost:3306 username@soic_machine.soic.indiana.edu
```

Once this is done, you can then access port 3306 on your local system and have that be tunnelled to port 3306 on the remote system. In the above example, you will need to replace `username` with your normal IU username.

If you wanted to use a different port number on your local computer than is used on the remote system, then you just need to adjust the `-L` argument accordingly. So, if you wanted to use port 12345 on your local computer to reach port 3306 on the remote system, you would use:

```
ssh -4 -N -L 12345:localhost:3306 username@soic_machine.soic.indiana.edu
```

Windows

There is a good tutorial for setting this up using Windows at [http://howto.ccs.neu.edu/howto/windows/ssh-port-tunneling-with-putty/](http://howto.ccs.neu.edu/howto/windows/ssh-port-tunneling-with-putty/)