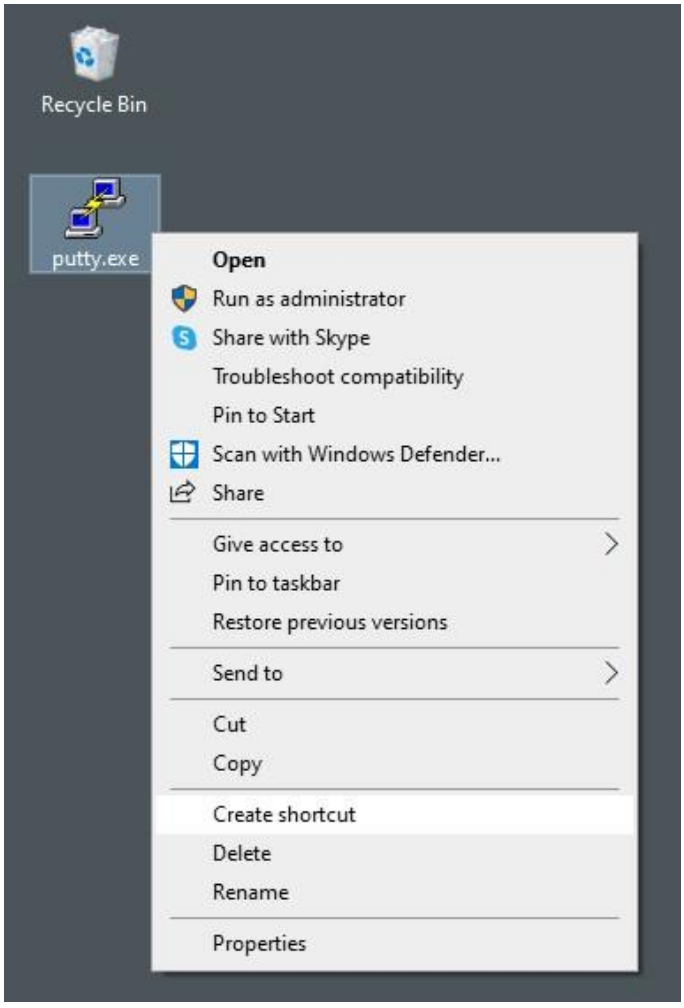
ECSC 325/425 SSH Tunnel Instructions

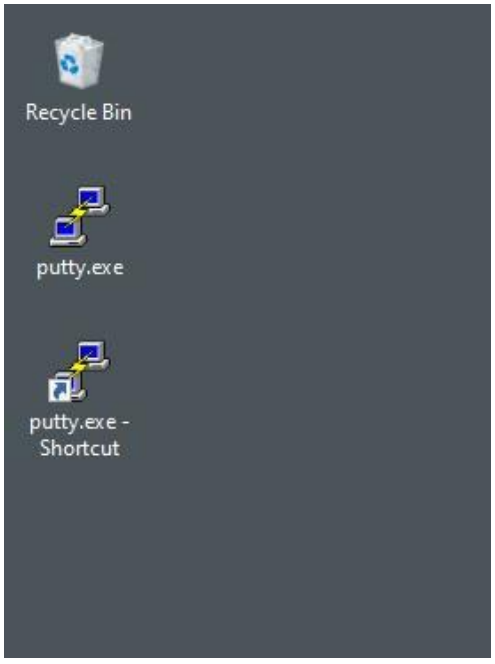
Using PuTTy on Windows

First, download PuTTy. For the purposes of this example, I will download to my desktop. Technically you can download it to anywhere, but to keep it simple in this example, I will use the desktop. I am just going to download the binary executable; the full installation package is not necessary.

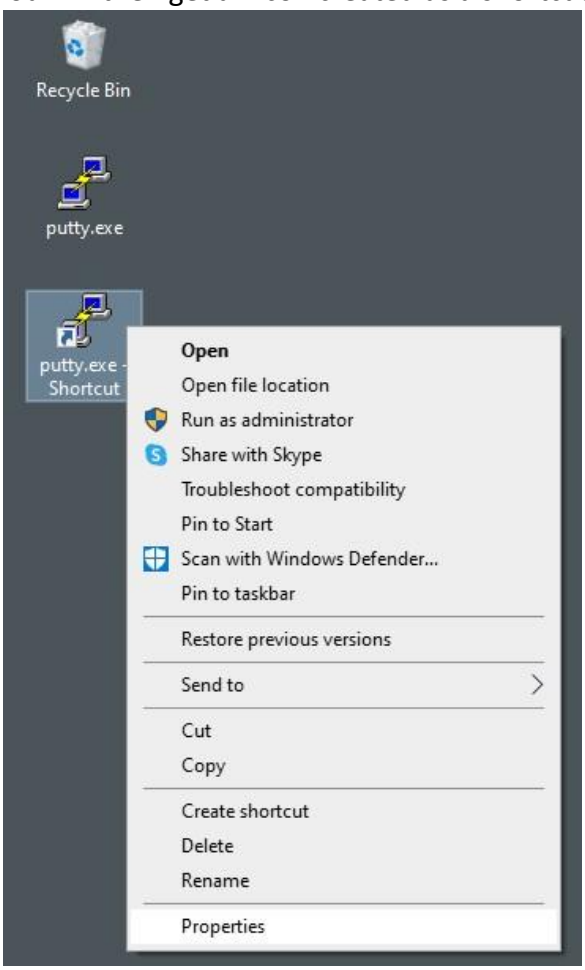
Putty on my desktop:

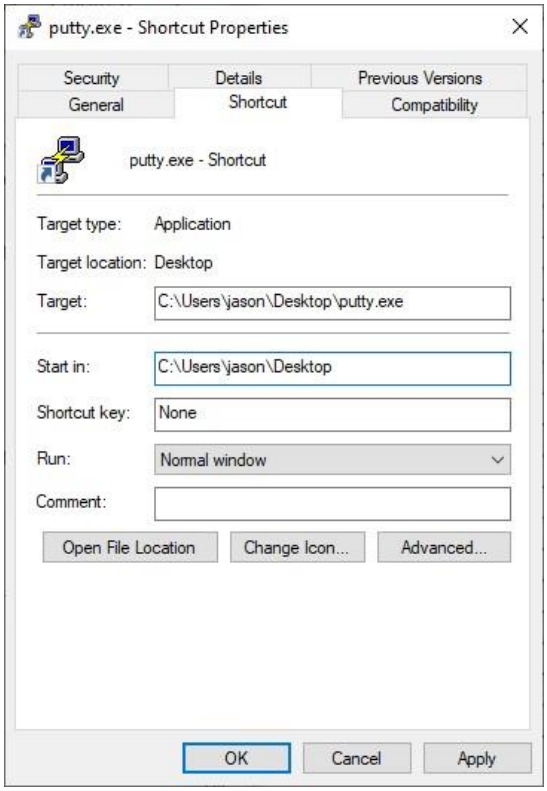
Next, create a shortcut to this new putty.exe program:





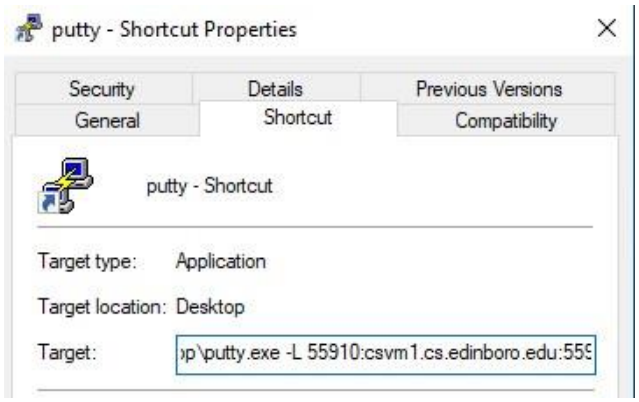
You will then get an icon created as a shortcut to putty. Right click on that shortcut icon, and go to properties:





In the Target location append the appropriate SSH parameters, similar to what I have discussed in class. In this case, I want to create a local port forward from port **55110** to server **csvm1.cs.edinboro.edu** port **5910**, and a dynamic port forward using port **55510** through server **cslab100.cs.edinboro.edu** on port 22:

putty.exe **-L 55910:csvm1.cs.edinboro.edu:55910 -D 55510 cslab100.cs.edinboro.edu -p 22**



The full string in my connection target box above is:
C:\Users\jason\Desktop\putty.exe -L 55910:csvm1.cs.edinboro.edu:55910 -D 55510 cslab100.cs.edinboro.edu -p 22

Make sure if you copy and paste my path, that you change the username and the ports for your usage! You should replace the last two digits of the ports to reflect your 2 digit number (mine is 10 in this example).

Using SSH on Mac, Linux, or newer versions of Windows

On Mac's and Linux machines (and newer versions of Windows; Windows 10 build 1809 and later, and Windows 11), you can simply open a command prompt and run the SSH command, without the need for Putty or any additional software.

Simply run a SSH command from the command line similar to the one below:

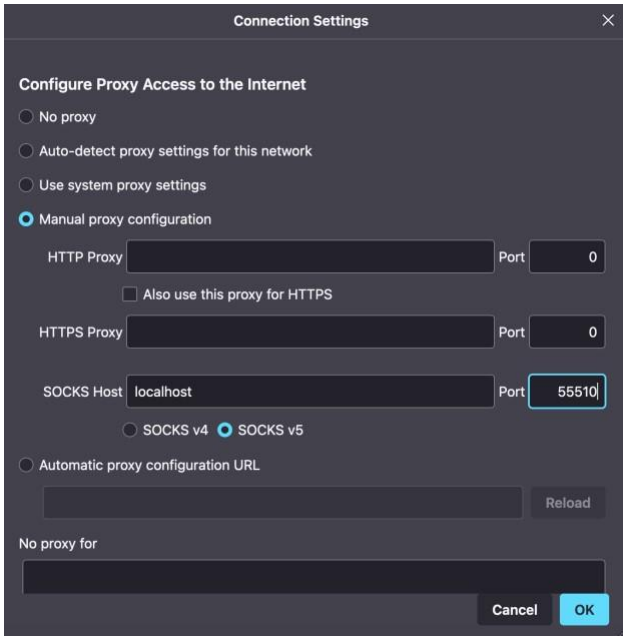
ssh -L 55110:csvm1.cs.edinboro.edu:5110 -D 55510 cslab100.cs.edinboro.edu -p 22

This command will have the same effect of the putty shortcut described above. You should replace the last two digits of the ports to reflect your 2 digit number (mine is 10 in this example).

Connecting to your websites

The Putty shortcut or SSH command you ran will connect you to the CS domain, where you can log in to cslab100.cs.edinboro.edu, and it will automatically open the Dynamic SOCKS Proxy on port 55510. We can then configure Firefox to use this port to make our connections.

Within Firefox, open the Options, select General, scroll to the bottom of the page and click the "Settings…" box in the Network Proxy section, then enable the proxy by selecting the "Manual proxy configuration option" and settings the SOCKS Host to "localhost" and the port you used in the SSH command above (55510 in this example):



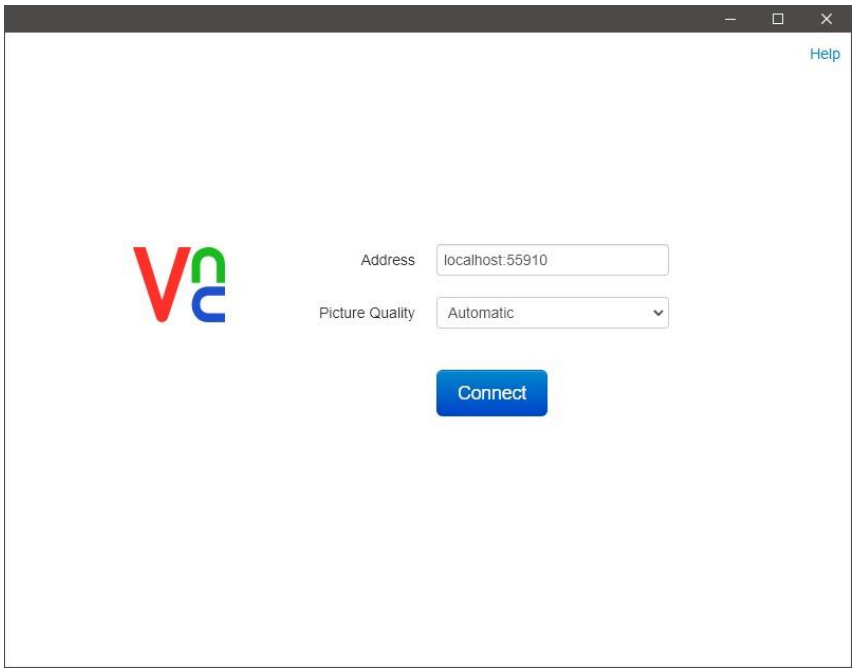
After configuring the proxy, you should be able to browse the web as if you were on the server you connected to with Putty/SSH! Open a website and give it a try!

You should replace the last two digits of the ports to reflect your 2 digit number (mine is 10 in this example).

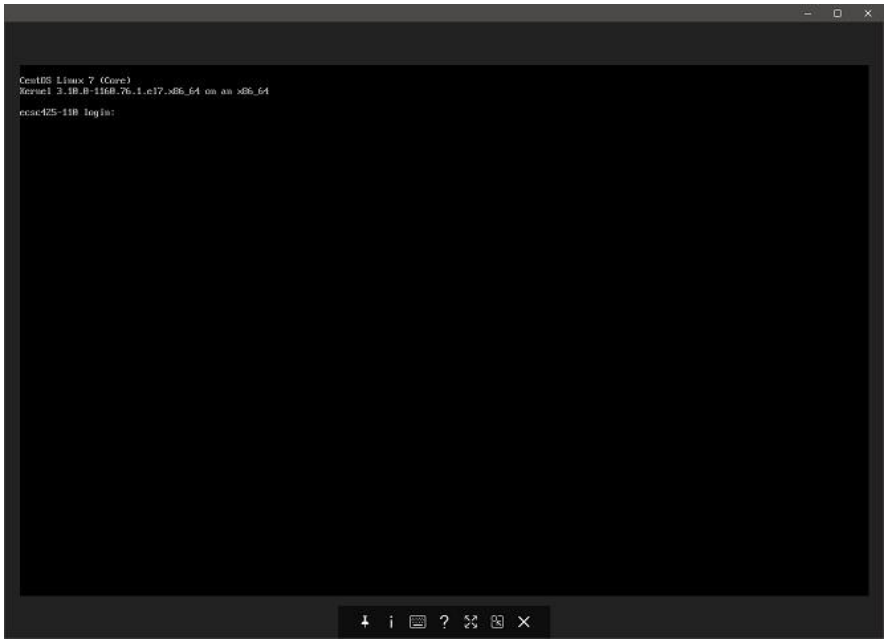
Out of Band Connections

The Putty shortcut or SSH command you ran above will connect you to the CS domain, where you can log in to cslab100.cs.edinboro.edu, and it will automatically create the port forward from local port 55910 to the OOB console connection on my VM at port 55910 on server csvm1.cs.edinboro.edu. You will want to replace the last two digits of the ports to reflect your specific port numbers! This uses our 2 digit numbers on the end (mine is 10 here). Please not that these OOB connections may not be configured for all classes!

Now, I can use the Virtual Network Computing (VNC) application to establish my console connection, through the tunnel I created, using localhost port 55910:



This VNC connection provides access to our VM Out of Band (OOB) console, where we can then log in to the console of our server with our root or user accounts:



Finally, below are some of the slides from class that discuss these topics.

SSH

SSH – Secure Shell

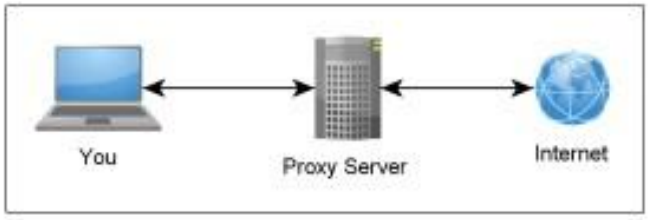
- Creates an encrypted terminal connection to a server shell
- Allows for other types of communication to be tunneled through connection
  - Tunneling insecure information through a secure tunnel creates a secure connection!
- In Windows, you can use the program Putty to SSH in to a server
- In Linux/MacOS, you can run the SSH command directly from the command line
- Command format:
  - ssh –L portX:host:portY –D portZ user@host
  - ssh –L 5901:hostname.fqdn.com:5901 –D 56565 user@hostname.tld
  - -L: forward local portX to remote host portY
  - -D: forward web requests through SOCKS proxy on portZ

SSH Tunnels

SSH Tunnels are secure and encrypted between endpoints!

Local port forwarding

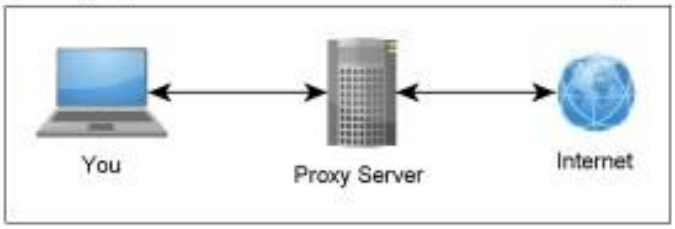
- Similar to port forwarding on a home router
- ssh –L source_port:destination_host:destination_port user@host.tld
  - Ex: `ssh -L 55110:newfilestore.cs.edinboro.edu:5110 user@cslab100.cs.edinboro.edu`
  - This command opens an encrypted tunnel between my local machine and ssh server cslab100.cs.edinboro.edu
  - Traffic sent to localhost port 55110 is redirected to newfilestore.cs.Edinboro.edu port 5110
  - You can do the same with a putty shortcut in Windows



SSH Tunnels

SOCKS (Dynamic) Proxy

- Allows for website proxying
- Can be used to allow connections around firewalls
  - Connect to your csci325 web server from anywhere through an intermediate host
- ssh –D port user@host.tld
  - Ex: `ssh -D 55555 jpatalon@cslab100.cs.edinboro.edu`
  - This command opens an encrypted tunnel between my local machine and ssh server cslab100.cs.edinboro.edu
  - Web SOCKS traffic is sent through port 55555 to server cslab100.cs.edinboro.edu, where it is then forwarded to the destination



SSH Tunnels

Combined SSH SOCKS proxy and local tunnels

- `ssh -D 55555 -L 33890:147.64.242.190:3389 jpatalon@cslab100.cs.edinboro.edu`
- This creates a tunnel through server cslab100.cs.Edinboro.edu
- This allows us to do both a local port forward and a dynamic port forward
- Connecting to localhost port 33890 is forwarded to 147.64.242.190 port 3389 through cslab100
- The connection is encrypted between the localhost and cslab100

SSH Tunnels

Multi-hop ssh tunnels

```
ssh -D 55555 -L 55556:127.0.0.1:55556 -L 55557:127.0.0.1:55557 jpatalon@cslab100.cs.edinboro.edu
ssh -D 55556 -L 55557:127.0.0.1:55557 user@csci425-101.math.cs.edinboro.edu
```