**SSH Secure Shell**

**SSH is an encrypted network protocol that provides an encrypted tunnel between the client and the server**

* Usually, TCP port 22

Natively, Linux and Mac come with SSH

Ssh -p 80 [root@demo.stationx.net](mailto:root@demo.stationx.net)

* -p changes the port ssh runs through
  + To get around firewall
* Root is the user that we’re connecting as
  + Would usually want to log in as standard user and su or sudo for root privileges
* Demo.stationx.net is the server, or can be an ip address

**Remote port forwarding**

Use the -R command

* Creates a local port available on the SSH server (only TCP)

In py3

* -m http.server 9999
* Creates a simple http server on port 9999

**Local Port Forwarding**

Ssh -L 8080:localhost:6666 root@demo.stationx.net

This creates an SSH server on port 22, which will provide an encrypted SSH tunnel to port 8080 which will then forward the connection on to port 6666 on the remote server (StationX)

So, connecting to https://localhost:8080 will actually enable you access to the StationX server locally on port 6666 via an SSH tunnel

This allows you to close external connections to the server on that port and only allows internal/local connections to the server on port 6666 via SSH

* Can block access to all ports that you don’t need people to access and then make them only accessible to you or anyone using the SSH remote login
* This gives you authentication and encryption

You can also forward to a specific destination

Ssh -f -L 8080:ftp.heanet.ie:21 root@demo.stationx.net -N

This would allow you to SSH into port 21 and access the [ftp.heanet.ie](ftp://ftp.heanet.ie) location

ftp localhost 8080

* Will take you ftp server via SSH server

You could want to do this if you wanted to access a certain service but the port wasn’t allowed

* i.e., ftp:21 isn’t allowed but ssh:22 is

**Dynamic Ports and SOCKS5 SSH Tunnelling**

Set up a local SOCKS5 proxy that creates an encrypted tunnel to the SSH server

A dynamic port allows you to use any protocol to get to any site (if the server allows it)

Ssh -fNn -D 8080 [root@demo.stationx.net](mailto:root@demo.stationx.net)

-f – go into background just before it executes command

N instructs SSH to not execute a command on the machine

n redirects standardin from devnull, prevents reading from standardin

* must be used when SSH is running in the background

-D specifies the local dynamic application-level port forwarding

To set up the SOCKS SSH proxy

* Settings 🡪 Network 🡪 Advanced 🡪 SOCKS Host 🡪 #add Localhost and the port (8080)