Michael Esposito

Network Defense 2

CTF-01

Table of Contents

[Host Enumeration 1](#_Toc509251237)

[Nmap output 1](#_Toc509251238)

[Host Exploitation 2](#_Toc509251239)

[Vulnerable Service - SMB 2](#_Toc509251240)

[Dumping Hashes 2](#_Toc509251241)

[Lateral Propagation 3](#_Toc509251242)

[Passing the Hash 3](#_Toc509251243)

[Conclusions 4](#_Toc509251244)

# Host Enumeration

First, network reconnaissance was performed. We were told that the target hosts were Windows Server 2003 instances running on the 192.168.90.0/24 network. I ran an nmap scan with host fingerprinting enabled in order to locate the Windows hosts. I discovered one host, 192.168.90.10, which had numerous open ports corresponding to important Windows-related services.

## Nmap output

The following command was executed on Kali (172.17.17.209).

nmap -A 192.168.90.0/24 -oN nmap.txt

It generated the following output:

PORT STATE SERVICE VERSION

53/tcp open domain Microsoft DNS

88/tcp open kerberos-sec Microsoft Windows Kerberos (server time: 2018-03-19 15:23:43Z)

135/tcp open msrpc Microsoft Windows RPC

139/tcp open netbios-ssn Microsoft Windows netbios-ssn

389/tcp open ldap Microsoft Windows Active Directory LDAP (Domain: snpp.local, Site: Default-First-Site)

445/tcp open microsoft-ds Windows Server 2003 3790 Service Pack 1 microsoft-ds

464/tcp open kpasswd5?

593/tcp open ncacn\_http Microsoft Windows RPC over HTTP 1.0

636/tcp open tcpwrapped

1025/tcp open msrpc Microsoft Windows RPC

1027/tcp open ncacn\_http Microsoft Windows RPC over HTTP 1.0

1037/tcp open msrpc Microsoft Windows RPC

1040/tcp open msrpc Microsoft Windows RPC

1048/tcp open msrpc Microsoft Windows RPC

3268/tcp open ldap Microsoft Windows Active Directory LDAP (Domain: snpp.local, Site: Default-First-Site)

3269/tcp open tcpwrapped

# Host Exploitation

## Vulnerable Service - SMB

With the list of open ports on one of the target Windows 2003 servers, I began to search for a discovered service with a publicly available exploit. SMB runs on port 445, and Google research indicated that this version of SMB has a metasploit module available.

<https://www.rapid7.com/db/modules/exploit/windows/smb/ms08_067_netapi>

The module “exploits a parsing flaw in the path canonicalization code of NetAPI32.dll through the Server Service. This module is capable of bypassing NX on some operating systems and service packs”. Since this module can potentially provide remote code execution and Windows 2003 Server is a proven target, I decided to give it a try. I loaded this module, entered the corresponding target information, and ran the exploit. The exploit was successful, and I was prompted with a meterpreter session.

## Dumping Hashes

The exploit was successful, and with the resulting meterpreter session I was able to dump the password hashes stored on the Windows server.

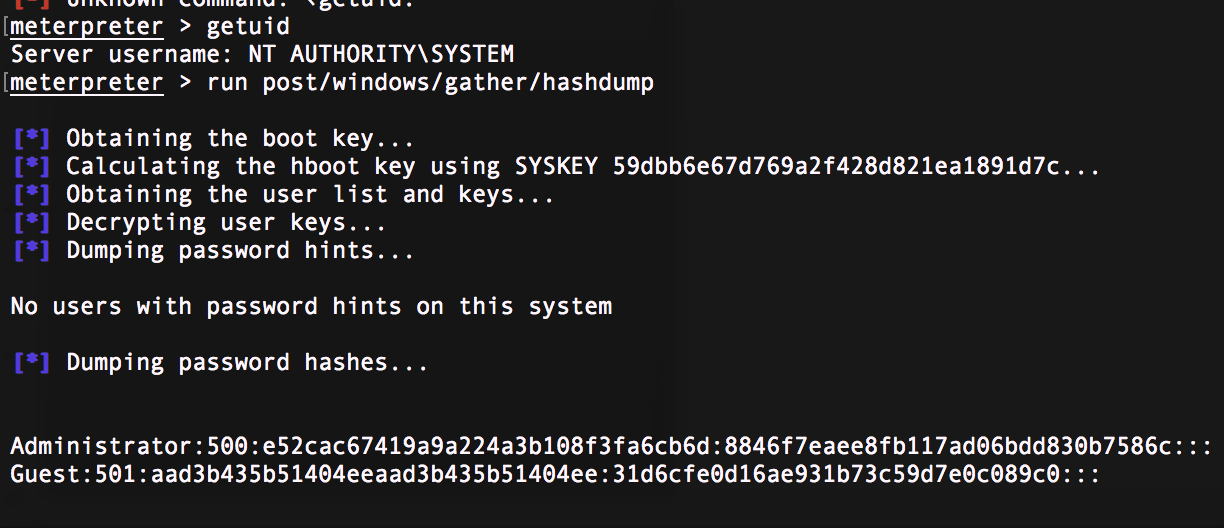


Figure : Dumping the password hashes

# Lateral Propagation

With access to two account hashes, I attempted to move laterally on the network. However, before doing so I needed the IP addresses of other Windows hosts on the network. Using the net and arp commands, I was able to enumerate the other Windows network devices.

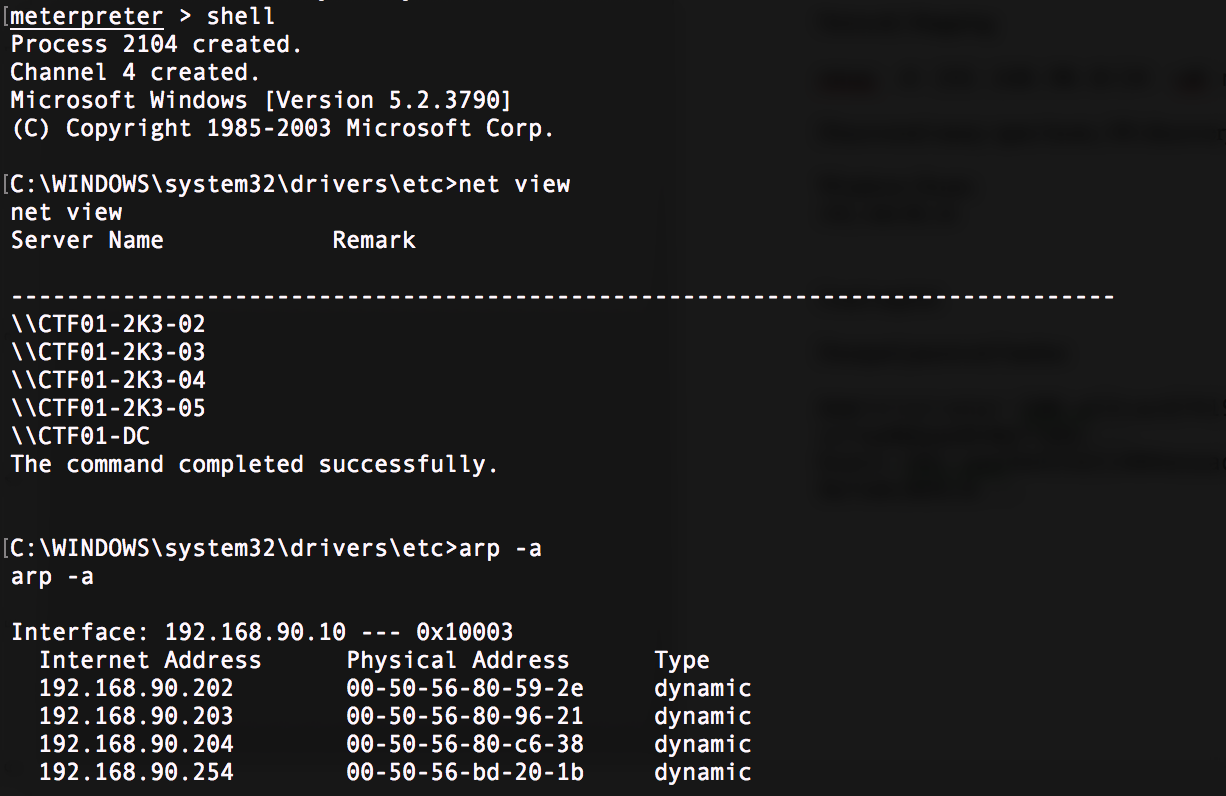


Figure : Enumerating Windows hosts

## Passing the Hash

Using the psexec module in metasploit, I was able to login to the other Windows 2003 servers on the network. Using the same Administrator hash on the following hosts, I dumped and recovered one more password hash from 192.168.90.2.2.

SUPPORT\_388945a0:1001:aad3b435b51404eeaad3b435b51404ee:b49b7750e2f7f2bdd3da9522f2b4feb4:::

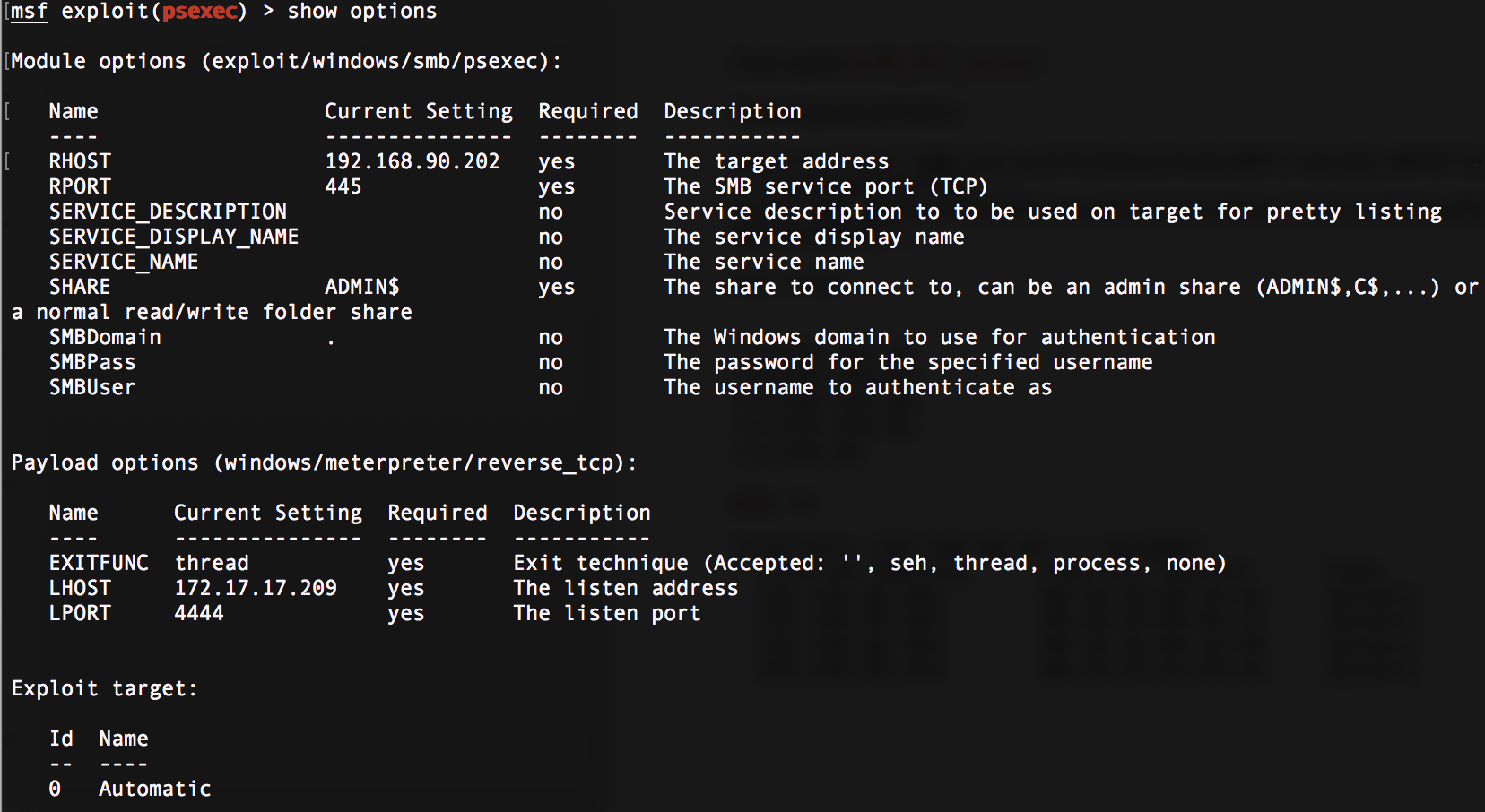


Figure : PSexec options

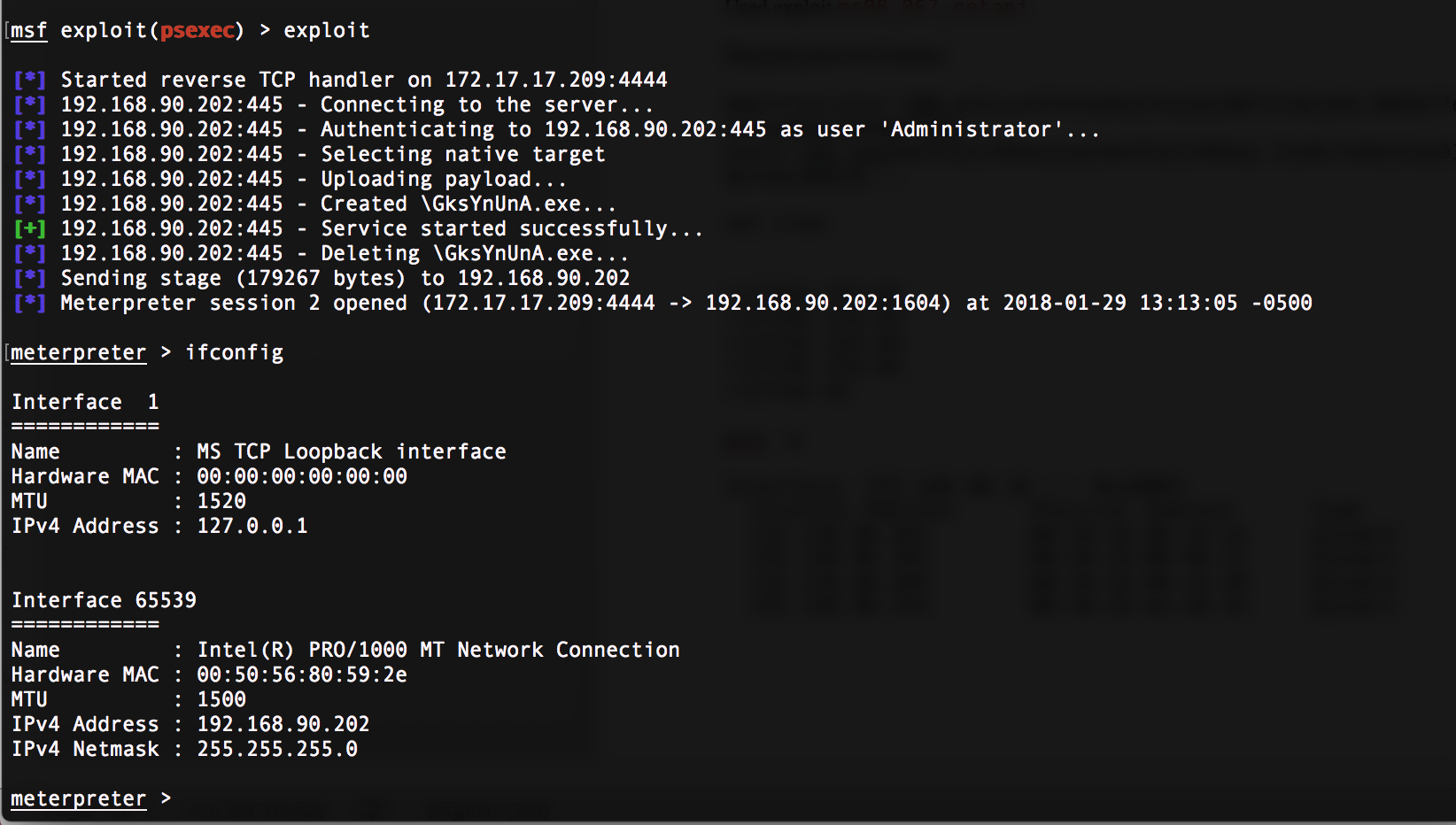


Figure : Meterpreter from successful pass the hash attempt on 192.16.90.202

# Conclusions

I learned how important it is to keep operating systems updated. I was not fully certain of the SMB version running on the original Windows server, but the first metasploit module I even tried ended up successfully exploiting the systems.