Lab4 Nmap

Lab Learning Objectives

- Use Nmap to perform various scan such as TCP, UDP, version and OS fingerprinting
- Use Nmap Scripting Engine
- Compare how Nmap behaves when NSE scripts are run with and without version scanning

Lab Setup

In this lab, you will use Kali Linux, Ubuntu Linux and Windows 10.

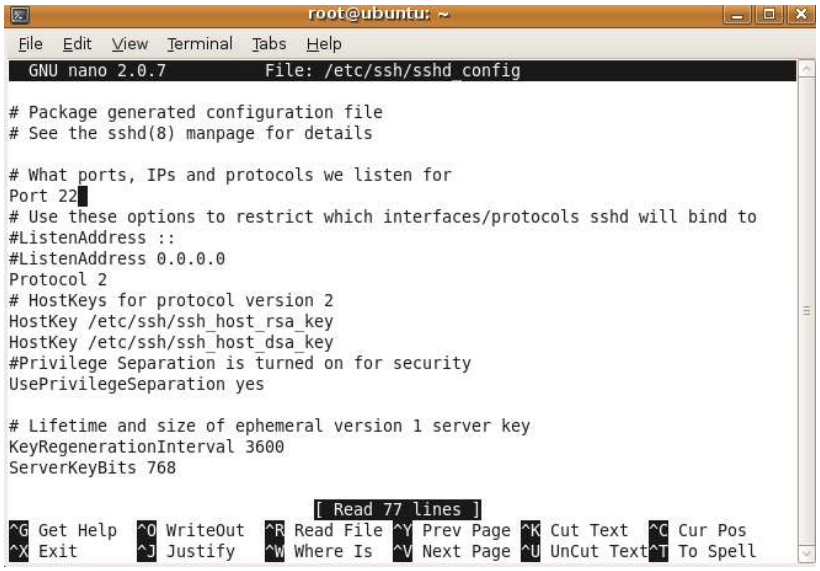
Lab Instructions

1. Bring up a terminal at Ubuntu Linux machine and su to the root

$ sudo su -

Enter georgia's password and you now have the root. Type

 # nano /etc/ssh/sshd_config



Change the line that says Port 22 to Port 23. Type CTRL-X to exit nano. Type y and hit enter to save the changes. Now, make your sshd reread its configuration file by sending it the HUP signal

 # killall -HUP sshd

Verify that your sshd is listening on TCP port 23 by typing

 # lsof -Pi | grep 23

The -i option indicates that we want to see network usage, whereas the –P modifier makes lsof display port numbers, not service names. If you see a line of output mentioning sshd and TCP 23, you are ready to go. Try to use the netstat command to accomplish the same task.

```
root@ubuntu: ~
File  Edit  View  Terminal  Tabs  Help
georgia@ubuntu:~$ sudo su -
[sudo] password for georgia:
root@ubuntu:~# nano /etc/ssh/sshd_config
root@ubuntu:~# killall -HUP sshd
root@ubuntu:~# netstat -nat | grep 23
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp6       0      0 :::23                   :::*                    LISTEN
root@ubuntu:~#
```

2. Move to Kali Linux machine and switch to the root account

$ sudo su -

Run Nmap against the Ubuntu VM

 # nmap -n Ubuntu_IP-Address

Namp reports that telnet is running at port 23. Remember that we moved ssh to a nonstandard port at 23 in step 1. Obvious, Nmap failed to catch it. This raises an interesting question. If a network admin to run a well known service say http from port 80 to a nonstandard port 3333. How can we find out that? In order to answer this question, we first need to understand how Nmap determine the service running at an open port.

```
root@kali: ~
File  Edit  View  Search  Terminal  Help
root@kali:~# nmap -n 153.91.153.151
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-11 10:56 EST
Nmap scan report for 153.91.153.151
Host is up (0.0053s latency).
Not shown: 993 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
23/tcp   open  telnet
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
2049/tcp open  nfs
MAC Address: 00:0C:29:1F:26:97 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.35 seconds
root@kali:~#
```

3. At Kali Linux command line, type (Make sure that you run gedit under kali not root. Otherwise, gedit will not open.)

$ sudo gedit /usr/share/nmap/nmap-services

```
Open ▾                    nmap-services                                Save
                          /usr/share/nmap
discard 9/tcp   0.003764        # sink null
discard 9/udp   0.015733        # sink null
unknown 10/tcp  0.000063
systat  11/tcp  0.000075        # Active Users
systat  11/udp  0.000577        # Active Users
unknown 12/tcp  0.000063
daytime 13/tcp  0.003927
daytime 13/udp  0.004827
unknown 14/tcp  0.000038
netstat 15/tcp  0.000038
unknown 16/tcp  0.000050
qotd    17/tcp  0.002346        # Quote of the Day
qotd    17/udp  0.009209        # Quote of the Day
msp     18/tcp  0.000000        # Message Send Protocol | Message Send Protocol (historic)
msp     18/udp  0.000610        # Message Send Protocol
chargen 19/tcp  0.002559        # ttytst source Character Generator | Character Generator
chargen 19/udp  0.015865        # ttytst source Character Generator
ftp-data        20/sctp 0.000000        # File Transfer [Default Data] | FTP
ftp-data        20/tcp  0.001079        # File Transfer [Default Data]
ftp-data        20/udp  0.001878        # File Transfer [Default Data]
ftp     21/sctp 0.000000        # File Transfer [Control] | File Transfer Protocol [Control]
ftp     21/tcp  0.197667        # File Transfer [Control]
ftp     21/udp  0.004844        # File Transfer [Control]
ssh     22/sctp 0.000000        # Secure Shell Login | The Secure Shell (SSH) Protocol
ssh     22/tcp  0.182286        # Secure Shell Login
ssh     22/udp  0.003905        # Secure Shell Login
telnet  23/tcp  0.221265
telnet  23/udp  0.006211
priv-mail       24/tcp  0.001154        # any private mail system
priv-mail       24/udp  0.000329        # any private mail system
smtp    25/tcp  0.131314        # Simple Mail Transfer
smtp    25/udp  0.001285        # Simple Mail Transfer
rsftp   26/tcp  0.007991        # RSFTP
nsw-fe  27/tcp  0.000138        # NSW User System FE
nsw-fe  27/udp  0.000395        # NSW User System FE
unknown 28/tcp  0.000050
msg-icp 29/tcp  0.000025        # MSG ICP
msg-icp 29/udp  0.000560        # MSG ICP
Plain Text ▾   Tab Width: 8 ▾   Ln 21, Col 36 ▾   INS
```

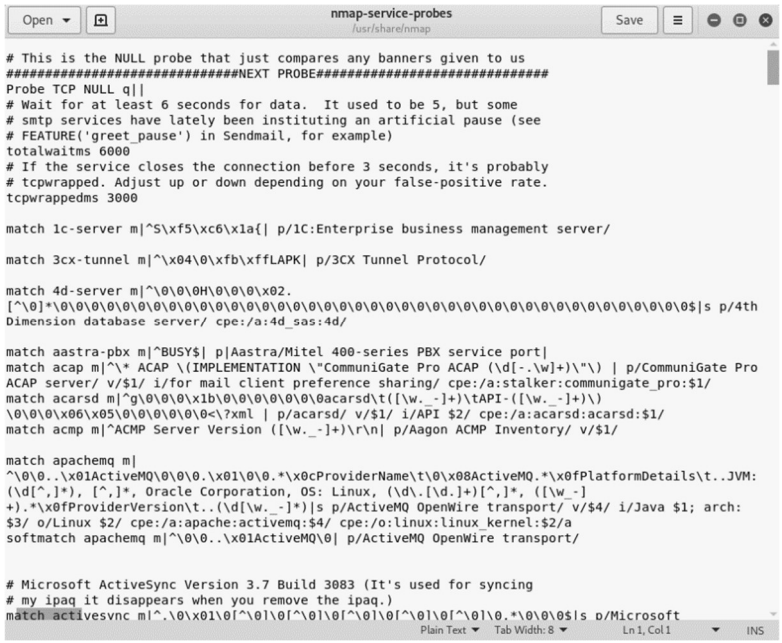
The format of this file includes the service name (for example, telnet), the associated port and protocol (for example, 23/tcp), the relative frequency with which the given port was discovered during Fyodor's widespread internet scanning research, and an optional comment. Close the file without saving.

In our previous case, Nmap performed a TCP stealthy scan (-sS) since we ran Nmap using the root privilege. Nmap discovers that TCP 23 was open but not realizing that it spoke ssh. Instead, it just looked up the "normal" service associated with that port from the nmap-services file, which is telnet. How to fix this problem? The answer is that we can use the version scan –sV.

4. Let's first see how Nmap conduct the version scan. At the Kali Linux terminal, run

$ sudo gedit /usr/share/nmap/nmap-service-probes

For the version scan -sV, Nmap bases its analysis of running services on the contents of a file called nmap-service-probes. In that file, lines that start with "Probe" indicate the messages to be sent to the target services, whereas lines that start with "match" indicate the response text to look for when determining the given service. Close the file without saving.



```
Open ▾   nmap-service-probes   /usr/share/nmap   Save

# This is the NULL probe that just compares any banners given to us
##############################NEXT PROBE##############################
Probe TCP NULL q||
# Wait for at least 6 seconds for data.  It used to be 5, but some
# smtp services have lately been instituting an artificial pause (see
# FEATURE('greet_pause') in Sendmail, for example)
totalwaitms 6000
# If the service closes the connection before 3 seconds, it's probably
# tcpwrapped. Adjust up or down depending on your false-positive rate.
tcpwrappedms 3000

match 1c-server m|^S\xf5\xc6\x1a{| p/1C:Enterprise business management server/

match 3cx-tunnel m|^\x04\0\xfb\xffLAPK| p/3CX Tunnel Protocol/

match 4d-server m|^\0\0\0H\0\0\0\x02.
[^\0]*\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0$|s p/4th
Dimension database server/ cpe:/a:4d_sas:4d/

match aastra-pbx m|^BUSY$| p|Aastra/Mitel 400-series PBX service port|
match acap m|^\* ACAP \(IMPLEMENTATION \"CommuniGate Pro ACAP (\d[-.\w]+)\"\) | p/CommuniGate Pro
ACAP server/ v/$1/ i/for mail client preference sharing/ cpe:/a:stalker:communigate_pro:$1/
match acarsd m|^g\0\0\0\x1b\0\0\0\0\0\0\0acarsd\t([\w._-]+)\tAPI-([\w._-]+)\)
\0\0\0\x06\x05\0\0\0\0\0<\?xml | p/acarsd/ v/$1/ i/API $2/ cpe:/a:acarsd:acarsd:$1/
match acmp m|^ACMP Server Version ([\w._-]+)\r\n| p/Aagon ACMP Inventory/ v/$1/

match apachemq m|
^\0\0..\x01ActiveMQ\0\0\0.\x01\0\0.*\x0cProviderName\t\0\x08ActiveMQ.*\x0fPlatformDetails\t..JVM:
(\d[^,]*), [^,]*, Oracle Corporation, OS: Linux, (\d\.[\d.]+)[^,]*, ([\w_-]
+).*\x0fProviderVersion\t..(\d[\w._-]*)|s p/ActiveMQ OpenWire transport/ v/$4/ i/Java $1; arch:
$3/ o/Linux $2/ cpe:/a:apache:activemq:$4/ cpe:/o:linux:linux_kernel:$2/a
softmatch apachemq m|^\0\0..\x01ActiveMQ\0| p/ActiveMQ OpenWire transport/


# Microsoft ActiveSync Version 3.7 Build 3083 (It's used for syncing
# my ipaq it disappears when you remove the ipaq.)
match activesync m|^.\0\x01\0[^\0]\0[^\0]\0[^\0]\0[^\0]\0[^\0]\0.*\0\0\0$|s p/Microsoft
Plain Text ▾   Tab Width: 8 ▾   Ln 1, Col 1   INS
```

5. Let's re-run the Nmap against the Ubuntu virtual machine with a version scan

 # nmap -n -sV Ubuntu IP-Address (Task01)

```
root@kali:~# gedit /usr/share/nmap/nmap-services
root@kali:~# gedit /usr/share/nmap/nmap-service-probes
root@kali:~# nmap -n -sV 153.91.153.151
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-11 10:48 EST
Nmap scan report for 153.91.153.151
Host is up (0.0042s latency).
Not shown: 993 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
23/tcp   open  ssh         OpenSSH 5.1p1 Debian 3ubuntu1 (Ubuntu Linux; protocol 2.0)
80/tcp   open  http        Apache httpd 2.2.9 ((Ubuntu) PHP/5.2.6-2ubuntu4.6 with Suhosin-Patch)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
2049/tcp open  nfs         2-4 (RPC #100003)
MAC Address: 00:0C:29:1F:26:97 (VMware)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

This time, Nmap correctly reports the service running at port 23 as ssh. Compared with the output from step 2, there is an extra column called Version shown in this output. It displays the version of each running service. We could use this information to conduct a vulnerability research to find out the potential exploits we can use to compromise the target system.

6. Move back to Ubuntu Linux machine to restore the ssh service back to port 22

 # nano /etc/ssh/sshd_config

Change the line that says Port 23 to Port 22. Type CTRL-X to exit nano. Type y and hit enter to save the changes. Now, make your sshd reread its configuration file by sending it the HUP signal

 # killall -HUP sshd

Verify that your sshd is listening on TCP port 22 by typing

 # netstat -nat | grep 22

7. Next, let's run the Nmap smb-os-discovery.nse script. This script will return target machine OS information including machine names which might be useful when we enumerate users by using tools such as sid2user.

At the Kali Linux command line, run

 # nmap -n --script=smb-os-discovery Windows 10 IP_address (Task02)

```
root@kali:~# nmap -n --script=smb-os-discovery 192.168.1.76
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-19 23:34 EST
Nmap scan report for 192.168.1.76
Host is up (0.00074s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
135/tcp open  msrpc
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:0C:29:EA:55:E4 (VMware)

Host script results:
| smb-os-discovery:
|   OS: Windows 10 Pro 17134 (Windows 10 Pro 6.3)
|   OS CPE: cpe:/o:microsoft:windows_10::-
|   Computer name: DESKTOP-OGNBOUP
|   NetBIOS computer name: DESKTOP-OGNBOUP\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2018-12-19T20:36:08-08:00

Nmap done: 1 IP address (1 host up) scanned in 1.94 seconds
```

From the output, we know that the Windows 10's name is DESKTOP-OGNBOUP. Also notice that, in addition to running the smb-os-discovery.nse script, Nmap also performed a port scan. Why? Because it needs to know which ports are open so that it can determine if the service the script tests is available.

8. Let's run the Nmap http-robots.txt.nse script. This script will pull the robots.txt file from target web server. The robots.txt file tells well behaved web crawlers to ignore given directories or pages on a website because they have information that the web owner does not want to be included in search engines. Let's try to pull the robots.txt from UCM web server. First, we need to get the IP address of UCM's web server.

 # ping www.ucmo.edu

We will not receive any response since the sever blocks ICMP echo request. However, we do get its IP address which is 153.91.1.10. What other methods can you use to get the web server's IP address? Now, we are ready to run the script

 # nmap -n --script=http-robots.txt 153.91.1.10 -p 80,443

We now see all the directories that are listed in the robots.txt file of the UCM website. Also notice that both ports 80 and 433 are open. Re-run the script by removing port 443 **(Task03).**

```
root@kali:~# nmap -n --script=http-robots.txt 153.91.1.10 -p 80,443
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-19 21:58 EST
Nmap scan report for 153.91.1.10
Host is up (0.039s latency).

PORT    STATE SERVICE
80/tcp  open  http
443/tcp open  https
| http-robots.txt: 24 disallowed entries (15 shown)
| /_training/ /_showcase/ /a-z.php /internal-resources/
| /editor-help/ /offices/university-relations/internal-resources/
| /college-of-arts-humanities-and-social-sciences/internal-resources/ /college-o
f-health-science-and-technology/school-of-technology/internal-resources/
| /college-of-health-science-and-technology/school-of-nutrition-kinesiology-psyc
hological-sciences/psychology/internal-resources/ /offices/accessibility-service
s/internal-resources/
| /offices/budget-and-planning/internal-resources/ /offices/graduate-and-interna
tional-student-services/internal-resources/
| /offices/student-financial-services/internal-resources/fac-staff/ /academics/u
cm-online/internal-resources/
|_/offices/accounting-services/internal-resources/

Nmap done: 1 IP address (1 host up) scanned in 0.80 seconds
```

9. Finally, let's conduct an OS fingerprinting on our Windows 10 machine and save the results into a xml file. At the Kali Linux terminal, run

 # nmap -n -O Windows 10 IP_Address -oX /tmp/Win10_Scan.xml

Also notice from the output that, besides the OS fingerprinting, Nmap also conducted a port scan for us.

```
root@kali:~# nmap -n -O 153.91.155.61
Starting Nmap 7.70 ( https://nmap.org ) at 2018-12-11 11:13 EST
Nmap scan report for 153.91.155.61
Host is up (0.0011s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: 00:0C:29:23:94:61 (VMware)
Device type: general purpose
Running: Microsoft Windows 10
OS CPE: cpe:/o:microsoft:windows_10:1703
OS details: Microsoft Windows 10 1703
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/
submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.86 seconds
```

Lab Report

- please include your name and 700# at the beginning of your report
- please upload your report to the Blackboard by the due date
- You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed
- only word or pdf format is acceptable

- you must show all the necessary commands associated with each task in order to receive credits
- your screenshots size must be appropriate to provide the visible details

Provide a report which includes the following item.

1. Provide screenshots for the Task01~03 (3 screenshots are needed)
2. Review the information about the whois-ip script at https://nmap.org/nsedoc/scripts/whois-ip.html. Run the script against an UCM IP address. Provide a screenshot showing the output of the script.

3. Review the information about the banner script at https://nmap.org/nsedoc/scripts/banner.html. Run the script against Ubuntu IP address. Provide a screenshot showing the output of the script.