Hacker Highschool
SECURITY AWARENESS FOR TEENS

LESSON 5
SYSTEM IDENTIFICATION

Hacker Highschool
SECURITY AWARENESS FOR TEENS

ISECOM
INSTITUTE FOR SECURITY AND OPEN METHODOLOGIES

"License for Use" Information

The following lessons and workbooks are open and publicly available under the following terms and conditions of ISECOM:

All works in the Hacker Highschool project are provided for non-commercial use with elementary school students, junior high school students, and high school students whether in a public institution, private institution, or a part of home-schooling. These materials may not be reproduced for sale in any form. The provision of any class, course, training, or camp with these materials for which a fee is charged is expressly forbidden without a license including college classes, university classes, trade-school classes, summer or computer camps, and similar. To purchase a license, visit the LICENSE section of the Hacker Highschool web page at www.hackerhighschool.org/license.

The HHS Project is a learning tool and as with any learning tool, the instruction is the influence of the instructor and not the tool. ISECOM cannot accept responsibility for how any information herein is applied or abused.

The HHS Project is an open community effort and if you find value in this project, we do ask you support us through the purchase of a license, a donation, or sponsorship.

Table of Contents

Contributors

Chuck Truett, ISECOM

Marta Barceló, ISECOM

Kim Truett, ISECOM

Pete Herzog, ISECOM

ISECOM
INSTITUTE FOR SECURITY AND OPEN METHODOLOGIES

5.0 Introduction

It is obvious that someone who sits down at the keyboard of your computer can gather information about it, including the operating system and the programs that are running, but it is also possible for someone to use a network connection to gather information about a remote computer. This lesson will describe some of the ways in which that information can be gathered. Knowing how this information is gathered will help you to ensure that your local computer is safe from these activities.

5.1 Identifying a Server

There are a number of useful sources on the Web which will allow you to collect information about domain names and IP addresses.

5.1.1 Identifying the Owner of a Domain

The first step in identifying a remote system is to look at the domain name or IP address. Using a *Whois* lookup, you can discover valuable information, including the identity of the owner of a domain and contact information, which may include addresses and phone numbers. Note that there are now a number of domain name registrars, and not all *whois* databases contain information for all domains. You may have to look at more that one *whois* database to find information on the domain that you are investigating.

5.1.2 Identifying the IP address of a Domain

There are a number of ways to determine the IP address of a domain. The address may be contained in the *whois* information or you may have to use a *DNS* or *Domain Name Service* lookup. (A web search engine will provide a number of resources for discovering IP addresses from domain names.)

Once you have the IP address, you can access the records of the various members of the *Number Resource Organization* (http://www.arin.net/ or http://www.ripe.net/), to gain information about how IP addresses are distributed. IP numbers are assigned to service providers and networks in large groups, and knowing which group an IP address is contained in, and who has the rights to that group, can be very useful. This can help you determine information about the server or service provider that a website uses.

Exercises:

Pick a valid domain name and use a *Whois* lookup to find out who owns that domain. dominio (http://www.whois.com -> "isecom.org"+Go -> Whois Lookup) What other information is available? When was the domain created? When will it expire? When was it last updated?

Find the IP address for this domain name. Using the *whois* lookups for the various members of the *Number Resource Organization* determine who this IP address has been assigned to. (Start with the *www.arin.net*, page, which also links to the other members of the NRO.) What is the range of the other numbers that have also been registered to this entity?

5.2 Identifying Services

Once you have established the owner and the IP address of a domain, then you can start to look for information about the server to which that domain refers.

5.2.1 Ping and TraceRoute

Now that you know who owns the domain, and who the IP number has been assigned to, you can check to see if the server that the website is on is actually active. The *ping* command will tell you if there is actually a computer associated with that domain or IP. The command

ping domain or

ping ipaddress

will tell you if there is an active computer at that address.

If the output of the *ping* command indicates that the packets sent were received, then you can assume that the server is active.

Another command, *tracert* (in Windows) or *traceroute* (in Linux) will show you the steps that information takes as it travels from your computer to the remote computer. Tracing the route that the packets take will sometimes give you additional information about the computers in the network with the computer that is the target of your trace. For example, computers will similar IP addresses will often be part of the same network.

Exercises:

Ping a valid website or IP address (ping www.isecom.org or ping 216.92.116.13). If you get a successful response, *ping* the next IP address. Did this produce a successful response?

Use *tracert* or *traceroute* to trace the route from your local computer to the IP address that you used in the previous exercise. How many steps does it take? Do any of the listed computers have similar IP addresses?

5.2.2 Banner Grabbing

The next step in identifying a remote system is to try to connect using telnet and FTP. The server programs for these services display text messages called banners. A banner may state clearly and precisely what server program is running. For example, when you connect to an anonymous FTP server, you might get the following message:

Connected to anon.server.

220 ProFTPD Server (Welcome . . .)

User (anon.server:(none)):

While the number 220 is an FTP code which indicates that the server is ready for a new user, the text message *ProFTPD Server* identifies the FTP server program that is running on the remote computer. Using a web search engine, you can learn what operating system the program runs on and other details about its requirements, capabilities, limitations, and flaws.

The primary flaw in the use of banner grabbing to gather information about a system is that clever system administrators can spoof banners. A banner that reads *NoneOfYourBusiness Server* is obviously misleading, but a Unix system with a banner that reads *WS_FTP Server* (a Windows-based FTP server) is going to complicate any intelligence gathering that may be done.

5.2.3 Identifying Services from Ports and Protocols

You can also determine what programs are running on a system by looking at what ports are open and what protocols are in use.

Start by looking at your own local computer. Go to a command line or shell prompt and run the *netstat* program using the *-a* (or all) switch:

```
netstat -a
```

The computer will display a list of open ports and some of the services that are using those ports:

```
Active Connections
```

```
Proto  Local Address                Foreign Address        State
TCP    YourComputer:microsoft-ds    YourComputer:0         LISTENING
TCP    YourComputer:1025            YourComputer:0         LISTENING
TCP    YourComputer:1030            YourComputer:0         LISTENING
TCP    YourComputer:5000            YourComputer:0         LISTENING
TCP    YourComputer:netbios-ssn     YourComputer:0         LISTENING
TCP    YourComputer:1110            216.239.57.147:http    TIME_WAIT
UDP    YourComputer:microsoft-ds    *:*
UDP    YourComputer:isakmp          *:*
UDP    YourComputer:1027            *:*
UDP    YourComputer:1034            *:*
UDP    YourComputer:1036            *:*
UDP    YourComputer:ntp             *:*
UDP    YourComputer:netbios-ns      *:*
UDP    YourComputer:netbios-dgm     *:*
```

From this you can see many of the programs and services that are running on your local computer – many of which you don't even realize are running.

Another program, called *fport*, provides information similar to that which *netstat* does, but it also details which programs are using the open ports and protocols. (Fport is available for free download from *www.foundstone.com*.)

Another program, called *nmap* (for *network mapper*), will more thoroughly probe your computer for open ports. When *nmap* is run, it will display a list of open ports and the services or protocols that use those ports. It may also be able to determine what operating system your computer is using. For example, if you run *nmap* on your local computer, you might see the following output:

```
Port  State Service
22/tcp      open  ssh
68/tcp      open  dhcpclient
139/tcp     open  netbios-ssn
445/tcp     open  microsoft-ds
Device type: general purpose
Running: Linux 2.4X|2.5.X
OS details: Linux Kernel 2.4.0 - 2.5.20
Uptime 1.024 days (since Sat Jul 4 12:15:48 2004)
```

Nmap is available on your Hacker Highschool or L. A. S. cd. It is also available for download from *www.insecure.org*.

Exercises:

Run netstat on your local computer, using the -a switch.

```
netstat -a
```

What ports are open? Using a web search engine, can you match these ports with the services that run on them? (This would be a good exercise to try at home, also, to see if your computer is running unnecessary – and potentially dangerous – services, such as FTP and telnet.)

Run *nmap*, using the -sS (for SYN Stealth scan), and -O (for guess operating system) switches and the IP address 127.0.0.1 as the target.

```
nmap -sS -O 127.0.0.1
```

The IP address 127.0.0.1 specifies the local host, or your local computer. (Note: this is different from the IP address that other computers on the internet use to communicate with yours; on any machine, the IP address 127.0.0.1 refers to the local computer) What open ports does nmap find? What services and programs are using these ports? Try running nmap while you have a web browser or telnet client open. Does this change the results?

5.3 System Fingerprinting

Now that you know how to identify a server and how to scan for open ports and use this information to determine what services are running, you can put this information together to *fingerprint* a remote system, establishing the most likely operating system and services that the remote computer is running.

5.3.1 Scanning Remote Computers

Using an IP address or a domain name other than 127.0.0.1 as an argument for *nmap* allows you to scan for open ports on remote computers. It doesn't mean that there will be open ports, or that you will find them, but it does allow you to try.

For example, imagine that you have been receiving a large amount of spam e-mails, and you want to discover information about the person who is sending you these e-mails. Looking at the headers of one of the e-mails, you see that many of the e-mails have originated from the same IP address: 256.92.116.13 (see **Lesson 9: E-mail Security** for more details on reading e-mail headers).

A *whois* lookup shows you that the address is part of a block assigned to a large ISP, but gives you no information regarding this particular IP address.

If you then use *nmap* to scan the computer at that address, you get the following results:

```
nmap -sS -O 256.92.116.13


Starting nmap 3.50 ( http://www.insecure.org/nmap ) at 2004-07-03 20:13
Eastern Daylight Time
Interesting ports on 256.92.116.13:
(The 1632 ports scanned but not shown below are in state: closed)
PORT      STATE    SERVICE
21/tcp    open     ftp
22/tcp    open     ssh
23/tcp    open     telnet
25/tcp    open     smtp
80/tcp    open     http
```

```
110/tcp   open     pop3
113/tcp   open     auth
135/tcp   filtered msrpc
136/tcp   filtered profile
137/tcp   filtered netbios-ns
138/tcp   filtered netbios-dgm
139/tcp   filtered netbios-ssn
143/tcp   open     imap
144/tcp   open     news
161/tcp   filtered snmp
306/tcp   open     unknown
443/tcp   open     https
445/tcp   filtered microsoft-ds
513/tcp   open     login
514/tcp   open     shell
No exact OS matches for host (If you know what OS is running on it, see
http://www.insecure.org/cgi-bin/nmap-submit.cgi).
TCP/IP fingerprint:
SInfo(V=3.50%P=i686-pc-windows-windows%D=7/3%Time=40E74EC0%O=21%C=1)
TSeq(Class=TR%IPID=RD%TS=1000HZ)
T1(Resp=Y%DF=Y%W=FFFF%ACK=S++%Flags=AS%Ops=MNWNNT)
T2(Resp=N)
T3(Resp=N)
T4(Resp=N)
T5(Resp=Y%DF=N%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=N)
T7(Resp=N)

Uptime 1.877 days (since Thu Jul 01 23:23:56 2004)

Nmap run completed -- 1 IP address (1 host up) scanned in 775.578 seconds
```

The ports marked as *filtered* are well-known as potentially vulnerable to attack, so it is not a surprise to find them listed as filtered. What is most interesting is that ports 21, 22 and 23 – for ftp, ssh and telnet – are all listed as open.

The last thing that *nmap* does is to try to identify the operating system that is running on the scanned computer. In this instance, the tests that *nmap* runs are inconclusive, however, since *nmap* does show that ftp and telnet services are both running, you can attempt to connect through each of those to see if there is a banner that will be broadcast.

When you connect through FTP you see a banner that says:

220 ftp316.pair.com NcFTPd Server (licensed copy) ready.

When you then connect through telnet, the computer displays a banner which says

FreeBSD/i386 (ttyp7)

A quick web search tells you that NcFTPd is a Unix program and that FreeBSD is a Unix-type operating system, so it is likely that the server is running a version of FreeBSD as its operating system. You can't be certain that this is accurate (banners can be spoofed), but you can accept this as a reasonable guess.

So, by using *nmap*, along with FTP and telnet, you have determined that the server which has been sending you spam runs a Unix-type operating system – probably FreeBSD – and is set up to send and receive a large variety of information, through a number of services including FTP, telnet, http, smtp and pop3.

Further Reading

Nmap: http://www.insecure.org/nmap/

More on Nmap:
http://www.networkmagazine.com/shared/article/showArticle.jhtml?articleId=8702942&classroom=

Fport:http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/proddesc/fport.htm

A number of site detailing ports and the services that use them:
http://www.chebucto.ns.ca/~rakerman/port-table.html
http://www.chebucto.ns.ca/~rakerman/port-table.html#IANA
http://www.iana.org/assignments/port-numbers
http://www.networksorcery.com/enp/protocol/ip/ports00000.htm

Various DNS lookups: http://www.dnsstuff.com/

Ping:http://www.freesoft.org/CIE/Topics/53.htm