Nmap Hackerworkshop

exercises

Henrik Lund Kramshoej
hlk@zencurity.com

September 4, 2018

Contents

Preface

This material is prepared for use in *ethical hacker workshop* and was prepared by Henrik Lund Kramshoej, http://www.zencurity.com . It describes the networking setup and applications for trainings and workshops where hands-on exercises are needed.

Further a presentation is used which is available as PDF from kramse@Github
Look for nmap-workshop-exercisesin the repo security-courses.

These exercises are expected to be performed in a training setting with network connected systems. The exercises use a number of tools which can be copied and reused after training. A lot is described about setting up your workstation in the repo

`https://github.com/kramse/kramse-labs`

Prerequisites

This material expect that participants have a working knowledge of TCP/IP from a user perspective. Basic concepts such as web site addresses and email should be known as well as IP-addresses and common protocols like DHCP.

Have fun and learn

Introduction to networking

IP - Internet protocol suite

It is extremely important to have a working knowledge about IP to implement secure and robust infrastructures. Knowing about the alternatives while doing implementation will allow the selection of the best features.

ISO/OSI reference model

A very famous model used for describing networking is the ISO/OSI model of networking which describes layering of network protocols in stacks.

This model divides the problem of communicating into layers which can then solve the problem as smaller individual problems and the solution later combined to provide networking.

Having layering has proven also in real life to be helpful, for instance replacing older hardware technologies with new and more efficient technologies without changing the upper layers.

In the picture the OSI reference model is shown along side with the Internet Protocol suite model which can also be considered to have different layers.

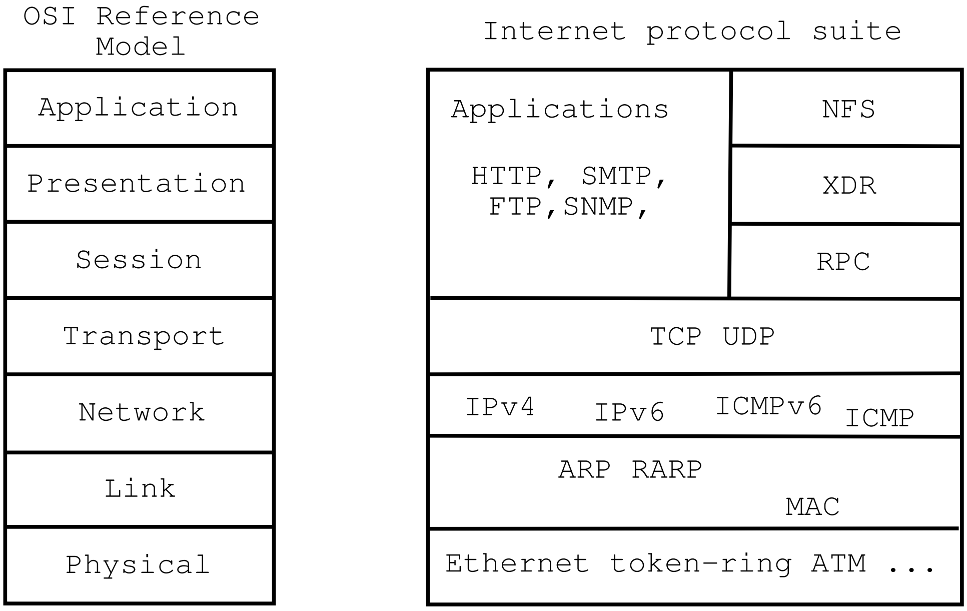


Figure 1: OSI og Internet Protocol suite

Exercise content

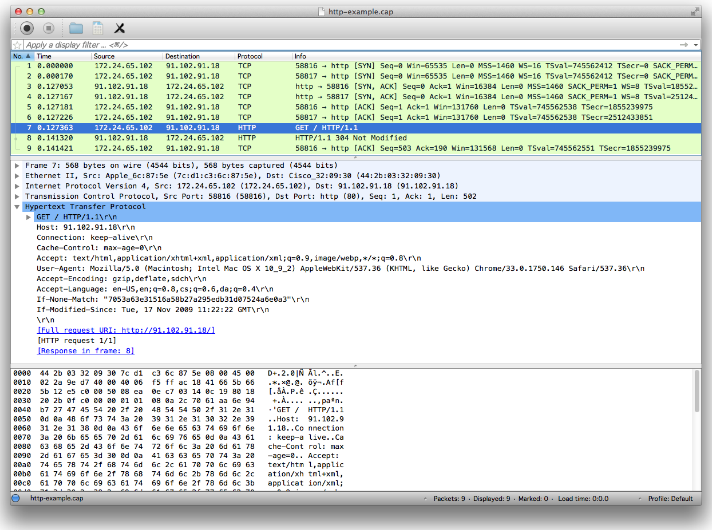
Most exercises follow the same procedure and has the following content:

- **Objective:** What is the exercise about, the objective
- **Purpose:** What is to be the expected outcome and goal of doing this exercise
- **Suggested method:** suggest a way to get started
- **Hints:** one or more hints and tips or even description how to do the actual exercises
- **Solution:** one possible solution is specified
- **Discussion:** Further things to note about the exercises, things to remember and discuss

Please note that the method and contents are similar to real life scenarios and does not detail every step of doing the exercises. Entering commands directly from a book only teaches typing, while the exercises are designed to help you become able to learn and actually research solutions.

Exercise 1

Wireshark installation



```
No. ▲ Time      Source         Destination    Protocol Info
  1 0.000000  172.24.65.102  91.102.91.18   TCP      58816 → http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM…
  2 0.000170  172.24.65.102  91.102.91.18   TCP      58817 → http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM…
  3 0.127053  91.102.91.18   172.24.65.102  TCP      http → 58816 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=18552…
  4 0.127167  91.102.91.18   172.24.65.102  TCP      http → 58817 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=25124…
  5 0.127181  172.24.65.102  91.102.91.18   TCP      58816 → http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=1855239975
  6 0.127226  172.24.65.102  91.102.91.18   TCP      58817 → http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=2512433851
  7 0.127363  172.24.65.102  91.102.91.18   HTTP     GET / HTTP/1.1
  8 0.141320  91.102.91.18   172.24.65.102  HTTP     HTTP/1.1 304 Not Modified
  9 0.141421  172.24.65.102  91.102.91.18   TCP      58816 → http [ACK] Seq=503 Ack=190 Win=131568 Len=0 TSval=745562551 TSecr=1855239975
```

```
▶ Frame 7: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits)
▶ Ethernet II, Src: Apple_6c:87:5e (7c:d1:c3:6c:87:5e), Dst: Cisco_32:09:30 (44:2b:03:32:09:30)
▶ Internet Protocol Version 4, Src: 172.24.65.102 (172.24.65.102), Dst: 91.102.91.18 (91.102.91.18)
▶ Transmission Control Protocol, Src Port: 58816 (58816), Dst Port: http (80), Seq: 1, Ack: 1, Len: 502
▼ Hypertext Transfer Protocol
  ▶ GET / HTTP/1.1\r\n
    Host: 91.102.91.18\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
    User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36\r\n
    Accept-Encoding: gzip,deflate,sdch\r\n
    Accept-Language: en-US,en;q=0.8,cs;q=0.6,da;q=0.4\r\n
    If-None-Match: "7053a63e31516a58b27a295edb31d07524a6e0a3"\r\n
    If-Modified-Since: Tue, 17 Nov 2009 11:22:22 GMT\r\n
    \r\n
    [Full request URI: http://91.102.91.18/]
    [HTTP request 1/1]
    [Response in frame: 8]
```

```
0000  44 2b 03 32 09 30 7c d1  c3 6c 87 5e 08 00 45 00   D+.2.0|Ñ Ãl.^..E.
0010  02 2a 9e d7 40 00 40 06  f5 ff ac 18 41 66 5b 66   .*.×@.@. õÿ¬.Af[f
0020  5b 12 e5 c0 00 50 08 ea  0e c7 03 14 0c 19 80 18   [.åÀ.P.ê .Ç......
0030  20 2b 0f c0 00 00 01 01  08 0a 2c 70 61 aa 6e 94    +.À.... ..,paªn.
0040  b7 27 47 45 54 20 2f 20  48 54 54 50 2f 31 2e 31   ·'GET /  HTTP/1.1
0050  0d 0a 48 6f 73 74 3a 20  39 31 2e 31 30 32 2e 39   ..Host:  91.102.9
0060  31 2e 31 38 0d 0a 43 6f  6e 6e 65 63 74 69 6f 6e   1.18..Co nnection
0070  3a 20 6b 65 65 70 2d 61  6c 69 76 65 0d 0a 43 61   : keep-a live..Ca
0080  63 68 65 2d 43 6f 6e 74  72 6f 6c 3a 20 6d 61 78   che-Cont rol: max
0090  2d 61 67 65 3d 30 0d 0a  41 63 63 65 70 74 3a 20   -age=0.. Accept:
00a0  74 65 78 74 2f 68 74 6d  6c 2c 61 70 70 6c 69 63   text/htm l,applic
00b0  61 74 69 6f 6e 2f 78 68  74 6d 6c 2b 78 6d 6c 2c   ation/xh tml+xml,
00c0  61 70 70 6c 69 63 61 74  69 6f 6e 2f 78 6d 6c 3b   applicat ion/xml;
```

http-example.cap · Packets: 9 · Displayed: 9 · Marked: 0 · Load time: 0:0.0 · Profile: Default

Objective:
Install the program Wireshark locally your workstation

If you already have Kali installed you have Wireshark. Done.

Purpose:
Installing Wireshark will allow you to analyse packets and protocols

Suggested method:
Download and install the program, either download from web server locally or from http://www.wireshark.org
Wireshark requires a packet capture library to be installed

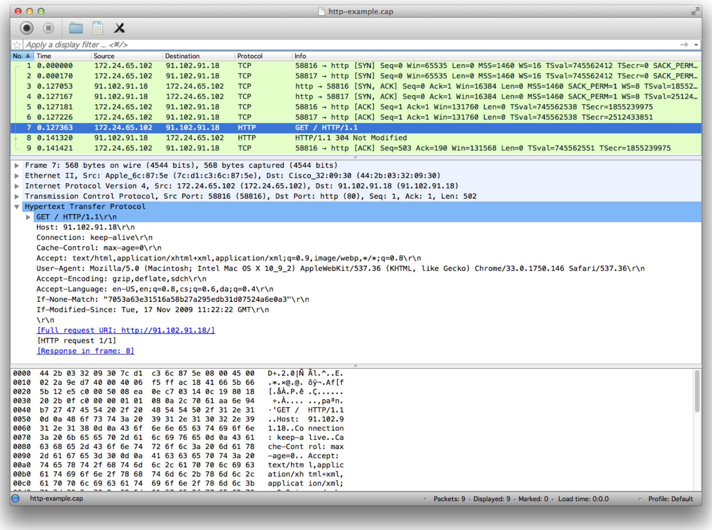
Hints:
PCAP is a packet capture library allowing you to read packets from the network. Wireshark is a graphical application to allow you to browse through traffic, packets and protocols.

Solution:
When Wireshark is installed sniff some packets, also see next exercise.

Discussion:
Wireshark is just an example other packet analyzers exist, some commercial and some open source like Wireshark

Exercise 2

Nmap installation


http-example.cap
Apply a display filter ... <⌘/>
http-example.cap
Packets: 9 · Displayed: 9 · Marked: 0 · Load time: 0:0.0
Profile: Default

```
No. ▲ Time      Source          Destination     Protocol  Info
    1 0.000000  172.24.65.102   91.102.91.18    TCP       58816 → http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM…
    2 0.000170  172.24.65.102   91.102.91.18    TCP       58817 → http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM…
    3 0.127053  91.102.91.18    172.24.65.102   TCP       http → 58816 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=18552…
    4 0.127167  91.102.91.18    172.24.65.102   TCP       http → 58817 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=25124…
    5 0.127181  172.24.65.102   91.102.91.18    TCP       58816 → http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=1855239975
    6 0.127226  172.24.65.102   91.102.91.18    TCP       58817 → http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=2512433851
    7 0.127363  172.24.65.102   91.102.91.18    HTTP      GET / HTTP/1.1
    8 0.141320  91.102.91.18    172.24.65.102   HTTP      HTTP/1.1 304 Not Modified
    9 0.141421  172.24.65.102   91.102.91.18    TCP       58816 → http [ACK] Seq=503 Ack=190 Win=131568 Len=0 TSval=745562551 TSecr=1855239975
```

```
▶ Frame 7: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits)
▶ Ethernet II, Src: Apple_6c:87:5e (7c:d1:c3:6c:87:5e), Dst: Cisco_32:09:30 (44:2b:03:32:09:30)
▶ Internet Protocol Version 4, Src: 172.24.65.102 (172.24.65.102), Dst: 91.102.91.18 (91.102.91.18)
▶ Transmission Control Protocol, Src Port: 58816 (58816), Dst Port: http (80), Seq: 1, Ack: 1, Len: 502
▼ Hypertext Transfer Protocol
  ▶ GET / HTTP/1.1\r\n
    Host: 91.102.91.18\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
    User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36\r\n
    Accept-Encoding: gzip,deflate,sdch\r\n
    Accept-Language: en-US,en;q=0.8,cs;q=0.6,da;q=0.4\r\n
    If-None-Match: "7053a63e31516a58b27a295edb31d07524a6e0a3"\r\n
    If-Modified-Since: Tue, 17 Nov 2009 11:22:22 GMT\r\n
    \r\n
    [Full request URI: http://91.102.91.18/]
    [HTTP request 1/1]
    [Response in frame: 8]
```

```
0000  44 2b 03 32 09 30 7c d1  c3 6c 87 5e 08 00 45 00   D+.2.0|Ñ Ãl.^..E.
0010  02 2a 9e d7 40 00 40 06  f5 ff ac 18 41 66 5b 66   .*.×@.@. õÿ¬.Af[f
0020  5b 12 e5 c0 00 50 08 ea  0e c7 03 14 0c 19 80 18   [.åÀ.P.ê .Ç......
0030  20 2b 0f c0 00 00 01 01  08 0a 2c 70 61 aa 6e 94    +.À.... ..,paªn.
0040  b7 27 47 45 54 20 2f 20  48 54 54 50 2f 31 2e 31   ·'GET /  HTTP/1.1
0050  0d 0a 48 6f 73 74 3a 20  39 31 2e 31 30 32 2e 39   ..Host:  91.102.9
0060  31 2e 31 38 0d 0a 43 6f  6e 6e 65 63 74 69 6f 6e   1.18..Co nnection
0070  3a 20 6b 65 65 70 2d 61  6c 69 76 65 0d 0a 43 61   : keep-a live..Ca
0080  63 68 65 2d 43 6f 6e 74  72 6f 6c 3a 20 6d 61 78   che-Cont rol: max
0090  2d 61 67 65 3d 30 0d 0a  41 63 63 65 70 74 3a 20   -age=0.. Accept: 
00a0  74 65 78 74 2f 68 74 6d  6c 2c 61 70 70 6c 69 63   text/htm l,applic
00b0  61 74 69 6f 6e 2f 78 68  74 6d 6c 2b 78 6d 6c 2c   ation/xh tml+xml,
00c0  61 70 70 6c 69 63 61 74  69 6f 6e 2f 78 6d 6c 3b   applicat ion/xml;
```

Objective:
Install the package of programs locally your workstation

If you already have Kali installed you have Wireshark. Done

Purpose:
Installing the Nmap package will allow you to use various tools on a daily basis

Suggested method:
Download and install the program, either download from web server locally or from http://www.nmap.org

Hints:
Nmap includes more than just the nmap tool. Nping is an awesome tool to test connectivity using multiple protocols

Solution:
When the package is installed we are ready for the next steps

Discussion:
There are other port scanners, some stateless and others stateful like Nmap

Exercise 3

Lookup Whois data

Objective:
Learn to use Whois databases

Purpose:
Knowing who to contact in case of problems on the internet is important, and also verifying before starting scanning is required.

Suggested method:
Use the website of RIPE NCC `https://www.ripe.net/` or their other site

`https://stat.ripe.net/`

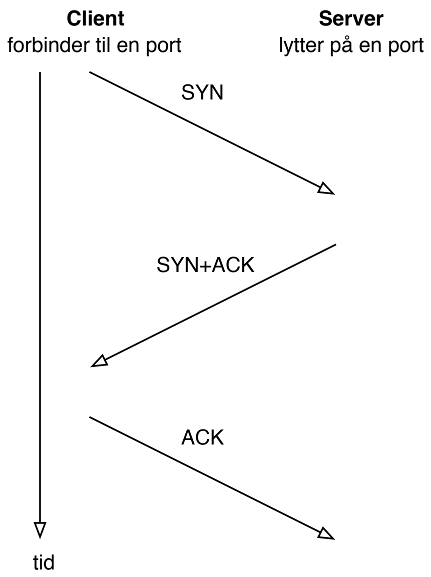
Hints:
Whois databases are distributed to Regional Internet Registries such as ARIN, AfriNIC, RIPE, LACNIC and APNIC.

Solution:
If you are using Linux or Mac you have a command line tool too:
Use the command whois with an IP address, `whois 185.129.60.130`.

Discussion:
The whois system was implemented after the Morris Worm affected the internet in November 1988, because it was realized that the internet had grown to a size that required more management.

Exercise 4

Sniffing network packets



Objective:
Sniff packets and dissect them using Wireshark

Purpose:
See real network traffic, also know that a lot of information is available and not encrypted.

Note the three way handshake between hosts

Suggested method:
Open Wireshark and start a capture
Then in another window execute the ping program while sniffing

or perform a Telnet connection while Wireshark sniff data

Hints:
When running on Linux the network cards are usually named eth0 for the first Ethernet and wlan0 for the first Wireless network card. In Windows the names of the network cards are long and if you cannot see which cards to use then try them one by one.

Solution:
When you have collected some packets you are done.

Discussion: Is it ethical to collect packets from an open wireless network?

Also note the TTL values in packets from different operating systems

Exercise 5

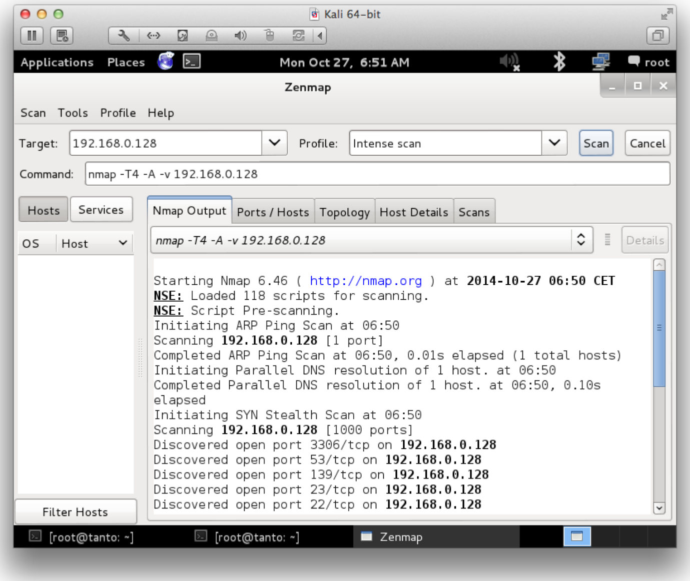
Nmap basic Quick Scan

Quick scan

Formål, * Se Zenmap * Opdage Tabs med hosts og services * Se farvekodning * Forskellen mellem 10.0.45.123/32 og 10.0.45.0/24 og 10.0.45.0/25 og 10.0.45.1-10 * Se de indbyggede profiler

Exercise 6

Discover active systems ping sweep



Target: 192.168.0.128

Profile: Intense scan

Command: nmap -T4 -A -v 192.168.0.128

```
nmap -T4 -A -v 192.168.0.128

Starting Nmap 6.46 ( http://nmap.org ) at 2014-10-27 06:50 CET
NSE: Loaded 118 scripts for scanning.
NSE: Script Pre-scanning.
Initiating ARP Ping Scan at 06:50
Scanning 192.168.0.128 [1 port]
Completed ARP Ping Scan at 06:50, 0.01s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 06:50
Completed Parallel DNS resolution of 1 host. at 06:50, 0.10s
elapsed
Initiating SYN Stealth Scan at 06:50
Scanning 192.168.0.128 [1000 ports]
Discovered open port 3306/tcp on 192.168.0.128
Discovered open port 53/tcp on 192.168.0.128
Discovered open port 139/tcp on 192.168.0.128
Discovered open port 23/tcp on 192.168.0.128
Discovered open port 22/tcp on 192.168.0.128
```

Objective:
Use nmap to discover active systems

Purpose:
Know how to use nmap to scan networks for active systems.

Suggested method:
Try different scans,

- Ping sweep to find active systems
- Port sweeps to find active systems with specific ports

Hints:
Try nmap in sweep mode

Solution:
Use the command below as examples:

- Ping sweep `nmap -sP 10.0.45.*`

- Port sweeps `nmap -p 80 10.0.45.*`

Discussion:

You can also use the graphical interface to nmap called Zenmap.

Exercise 7

Execute nmap TCP and UDP port scan

Objective:
Use nmap to discover open ports on active systems

Purpose:
Finding open ports will allow you to find vulnerabilities on these ports.

Suggested method:
Use `nmap -p 1-1024 server` to scan the first 1024 TCP ports

Try to use `nmap -sU` to scan using UDP ports, not really possible if a firewall is in place.

If a firewall blocks ICMP you might need to add `-P0` or even `-PN` to make nmap scan even if there are no Ping responses

Hints:
Sample command: `nmap -P0 -sU -p1-1024 server` UDP port scanning 1024 ports without doing a Ping first

Solution:
Discover some active systems and you are done.

Discussion:
There is a lot of documentation about the nmap portscanner, even a book by the author of nmap. Make sure to visit http://www.nmap.org

TCP and UDP is very different when scanning. TCP is connection/flow oriented and requires a handshake which is very easy to identify. UDP does not have a handshake and most applications will not respond to probes from nmap. If there is no firewall the operating system will respond to UDP probes on closed ports - and the ones that do not respond must be open.

When doing UDP scan on the internet you will almost never get a response, so you cannot tell open (not responding services) from blocked ports (firewall drop packets). Instead try using specific service programs for the services, sample program could be `nsping` which sends DNS packets, and will often get a response from a DNS server running on UDP port 53.

Exercise 8

Perform nmap OS detection

Objective:
Use nmap OS detection and see if you can guess the devices on the network

Purpose:
Getting the operating system of a system will allow you to focus your next attacks.

Suggested method:
Look at the list of active systems, or do a ping sweep.

Then add the OS detection using the option `-O`

Hints:
Use the manual page

The nmap can send a lot of packets that will get different responses, depending on the operating system.

Solution:
Use a command like `nmap -O -p1-100 10.0.45.45`

Discussion:
nmap OS detection is not a full proof way of knowing the actual operating system, but in most cases in can detect the family and in some cases it can identify the exact patch level of the system.

Another tool which does the same is Xprobe.

Exercise 9

Perform nmap service scan

Objective:
Use more advanced features in nmap to discover services.

Purpose:
Getting more intimate with the system will allow more precise discovery of the vulnerabilities and also allow you to select the next tools to run.

Suggested method:
Use `nmap -A` option for enabling service detection

Hints:
Look into the manual page of nmap or the web site book about nmap scanning

Solution:
Run nmap and get results.

Discussion:

Some services will show software versions allowing an attacker easy lookup at web sites to known vulnerabilities and often exploits that will have a high probability of success.

Make sure you know the difference between a vulnerability which is discovered, but not really there, a false positive, and a vulnerability not found due to limitations in the testing tool/method, a false negative.

A sample false positive might be reporting that a Windows server has a vulnerability that you know only to exist in Unix systems.

Exercise 10

Lav en specifik Nmap profil

Prøv at scanne en række porte, 1-1024 et antal udvalgte, 22,23,80,443,8080

Exercise 11

E) UDP scan på LAN

Lav UDP scan på lokalnet, closed = ICMP aktivt svar, open = open|filtered

Bonus: Hvorfor virker det ikke ude i verden, firewalls Koblet med Pingscan, hvad siger det så

Efter The six port states recognized by Nmap

Exercise 12

Fuld scan

, hvad betyder det

Formål * Vise at Nmap / portscan er en process, mere end

1) quick scan 2) Advanced, alle porte TCP -p 1-65535 med -A 3) -p 100 -sU UDP scan 4) … Husk -P0 -Pn: Treat all hosts as online – skip host discovery

5) Specific source ports -g -g/–source-port <portnum>: Use given port number FTP 20, DNS 53

Hvornår bruges fuld scan versus enkelte scans på tværs af subnets

Bonus -iL input liste af prefixes -iL <inputfilename>: Input from list of hosts/networks

Vi designer sammen en process og taler om resultaterne

Exercise 13

Hvad indeholder Nmap idag

Vi gennemgår seneste release notes, 7.70 pt. http://seclists.org/nmap-announce/2018/0

Bonus tal om hvordan man submitter fingerprints til Nmap

Exercise 14

Reporting HTML

Få resultaterne pænt ud af Nmap

Brug option -oA - alle formater

Formål * Forstå fordele ved GNMAP og XML

xsltproc xml html

–stylesheet <path/URL>: XSL stylesheet to transform XML output to HTML –webxml: Reference stylesheet from Nmap.Org for more portable XML Ndiff

Exercise 15

Find systems with SNMP

Objective:
Use snmpwalk to research SNMP systems

Purpose:
Learn that gathering information can help an attacker.

Suggested method:
Log into the Unix server provided and run snmpwalk which is using UDP port 161.

Hints:
We are running in a LAN environment with less firewalls, so doing nmap UDP scan is possible.

When discovering an IP then use the `snmpwalk` program to show a lot of information.

Solution:

- Use the command `snmpwalk -v 2c -c public 10.0.45.34 | less`

The command less will show output one screen at a time.

Discussion:
In real networks SNMP is being used a lot, but new equipment is starting NOT to allow access using the community string public.

Exercise 16

Nmap Scripting Engine NSE scripts

Lav en profil til Heartbleed, og SSL version 2 og version 3 scan Formål * Der dukker hele tiden systemer op med de gamle SSL versioner :-(* Forstå hvad et NSE script er * Forstå hvad NSE scripts er, banner grabbing, Advanced scan

Eksempler på real-life scenarioer nmap -sU -p 53 –script=dns-recursion -iL targets -oA dns-recursive

```
bgpscan.gnmap:# Nmap 7.70 scan initiated Thu May 31 12:22:49 2018 as: nmap -A -p 179 -
oA bgpscan -iL targets
dns-recursive.gnmap:# Nmap 7.70 scan initiated Mon Sep  3 12:24:48 2018 as: nmap -sU -
p 53 --script=dns-recursion -iL targets -oA dns-recursive
php-scan.gnmap:# Nmap 7.70 scan initiated Mon Jul 30 15:17:50 2018 as: nmap -sV --script=http-
php-version -p80,443 -oA php-scan -iL targets
scan-vtep-tcp.gnmap:# Nmap 7.70 scan initiated Wed May 16 12:05:03 2018 as: nmap -A -
p 1-65535 -oA scan-vtep-tcp 109.105.96.77 109.105.96.78
snmp-109.105.96.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-109.105.96.0 109.105.96.0/19
snmp-185.174.116.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-185.174.116.0 185.174.116.0/22
snmp-192.36.171.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-192.36.171.0 192.36.171.0/24
snmp-193.10.252.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-193.10.252.0 193.10.252.0/24
snmp-193.10.254.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-193.10.254.0 193.10.254.0/24
snmp-193.10.68.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-193.10.68.0 193.10.68.0/24
snmp-193.10.94.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-193.10.94.0 193.10.94.0/24
snmp-193.10.95.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-193.10.95.0 193.10.95.0/24
snmp-193.11.3.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:24 2018 as: nmap -sV -
A -p 161 -sU --script=snmp-info -oA snmp-193.11.3.0 193.11.3.0/24
snmp-194.68.13.0.gnmap:# Nmap 7.70 scan initiated Thu Jun 28 09:34:26 2018 as: nmap -
sV -A -p 161 -sU --script=snmp-info -oA snmp-194.68.13.0 194.68.13.0/24
snmpscan.gnmap:# Nmap 7.70 scan initiated Thu May 24 14:16:12 2018 as: nmap -sU -p 161 -
oA snmpscan --script=snmp-interfaces -iL targets
sshscan.gnmap:# Nmap 7.70 scan initiated Thu May 24 13:30:02 2018 as: nmap -A -p 22 -
oA sshscan -iL targets
vncscan.gnmap:# Nmap 7.70 scan initiated Thu May 24 13:52:56 2018 as: nmap -A -p 5900-
5905 -oA vncscan -iL targets
webscan-110.gnmap:# Nmap 7.70 scan initiated Wed May 16 13:16:37 2018 as: nmap -A -oA webscan-
```

```
110 109.105.110.0/24
```

Bonus Sjove scripts / andre scripts sudo nmap –traceroute –script traceroute-geolocation.nse -p 80 hackertarget.com

nmap –script http-enum 192.168.10.55

Til dem som har Windows med :-D og glemt at slå firewall til nmap -p 445 –script smb-os-discovery 192.168.1.0/24

Exercise 17

Kopier et NSE script ind

Find NSE script, kopier dette ind, kør Nmap med dette, enten via CLI eller Zenmap

nmap –script ”http-*”

nmap –script ”default or safe” This is functionally equivalent to nmap –script ”default,safe”. It loads all scripts that are in the default category or the safe category or both.

nmap –script ”default and safe” Loads those scripts that are in both the default and safe categories.

nmap -script-help http-vuln-cve2013-0156.nse

Exercise 18

Skriv vores eget NSE script

Vi starter en process med et bestemt indhold, laver et NSE script til detektion af dette

Måske bruge DNS signeret som eksempel? Findes der til den der LDAP UDP ting?

Lettere at modificere et eksisterende måske

https://nmap.org/nsedoc/scripts/http-default-accounts.html

–script-trace

https://nmap.org/book/nse-usage.html

```
  nmap -sC --script-args 'user=foo,pass=",=bar",paths=/admin,/cgi-bin,xmpp-info.server_name=lo
```

Eksempel, lav vores egen der leder efter bestemt version af OpenSSH, ref https://isc.sans.edu/forums/diary/OpenSSH+user+enumeration+CVE201815473/24004/ https://blog.nviso.be/2018/08/
user-enumeration-vulnerability-a-close-look/

og

CVE-2018-15919 en anden, som der ikke er fix til pt.

Eller lede efter PHP 5.5, som havde EOL 21 Jul 2016 !

https://nmap.org/nsedoc/scripts/http-php-version.html

Exercise 19

Try the bind-version shell script

Objective: Try to use a shell script to automate lookups

Purpose:
When doing actual security testing you should automate as much as possible.

Suggested method: Login to the Unix server provided and run the bind-version script

Hints: Unix files with `#!` as the first line will be executed using the command specified.

Solution:
Run the script provided

Discussion: The script only does a few DNS lookups, but more elaborate scripts are being used daily by administrators, security consultants and hackers.

The script available on the system is:

```
#! /bin/sh
# Try to get version info from BIND server
# many ways to do it
# nslookup -q=txt -class=CHAOS version.bind. 0
# dig @$* version.bind chaos txt
PROGRAM=`basename $0`
TARGET=$1

if [ $# -ne 1 ]; then
   echo "get name server version, need a target! "
   echo "Usage: $0 target"
   echo "example $0 10.1.2.3"
   exit 0
fi

# using dig
dig @$1 hostname.bind chaos txt
dig @$1 ID.SERVER chaos txt
dig @$1 version.bind chaos txt
dig @$1 authors.bind chaos txt
```

Exercise 20

Try the dns-timecheck Perl program

Objective: Try to use a Perl script to communicate with a binary protocol

Purpose:
See that programming languages such as Perl often include a lot of libraries which allow efficient implementation of ideas.

Suggested method: Login to the Unix server provided and run the dns-timecheck script

Hints: Perl can be a bit difficult to read, but a lot of tutorials exist

Solution:

Discussion: While Perl has been around for lots of years it seems that security tools are often implemented using newer languages like Ruby and Python

The script available on the system is:

```
#!/usr/bin/perl
# modified from original by Henrik Kramshoej, hlk@kramse.dk
# 2004-08-19
#
# Original from:
# http://www.rfc.se/fpdns/timecheck.html

use Net::DNS;

my $resolver = Net::DNS::Resolver->new;
$resolver->nameservers($ARGV[0]);

my $query = Net::DNS::Packet->new;
$query->sign_tsig("n","test");

my $response = $resolver->send($query);
foreach my $rr ($response->additional) {
 print "localtime vs nameserver $ARGV[0] time difference: ";
 print$rr->time_signed - time() if $rr->type eq "TSIG";
}
```

Exercise 21

Nping øvelser

Vigtige!

Exercise 22

Bonus: Try Nmap from Metasploit

Exercise 23

Bonus: Try masscan

https://github.com/robertdavidgraham/masscan

This is the fastest Internet port scanner. It can scan the entire Internet in under 6 minutes, transmitting 10 million packets per second.

Formål: * Tale om stateless vs stateful scanning

Exercise 24

Bonus: Network scripting using ncat

Objective:
Learn how to use the netcat program for scripting

Purpose:
Learn that a lot of protocols on the internet are easy read and create tools for.

Suggested method:
Login to the Unix server - look at the manualen `man nc`. Then create a textfile named headh.sh using this content

```
#! /bin/sh
# get HEAD from Webserver
cat | nc $1 $2 << EOF
HEAD / HTTP/1.0

EOF
```

Then use the command `chmod +x head.sh` to make it executable and run it

Hints:
The netcat program is a swiss army-knife for network data, and allows you to forward data to various ports and connect programs.

Solution:
Run the program: `./head.sh www.pentest.dk 80`

Discussion:

Sometime the program will seem to hang, use ctrl-c to break it.

Appendix A

Host information

- You should note the IP-addresses used for servers and devices
- The web server for installing programs:
  http://   .   .   .   /public/windows/
- Server used for team login: .   .   .
  Available usernames: team1, team2, ... team10 password: `team`
- You can obtain root access using: `sudo -s`

Available servers and devices:

- IP: .   .   .   - OpenBSD router
- IP: .   .   .   - Your laptop
- IP: .   .   .   - Your laptop VM
- IP: .   .   .   -
- IP: .   .   .   -