

# Internet Self Defence

## workshop

Henrik Lund Kramshøj

hlk@solido.net

5. oktober 2012



# Indhold

|           |  |           |
|-----------|--|-----------|
| <b>1</b>  | <b>Installation af alternativ browser</b>      | <b>3</b>  |
| <b>2</b>  | <b>Installation af Thunderbird</b>             | <b>4</b>  |
| <b>3</b>  | <b>Installation af GPG GNU Privacy Guard</b>   | <b>5</b>  |
| <b>4</b>  | <b>Installation af Enigmail plugin</b>         | <b>6</b>  |
| <b>5</b>  | <b>Installation af Truecrypt</b>               | <b>7</b>  |
| <b>6</b>  | <b>Lav en PGP-kompatibel nøgle</b>             | <b>8</b>  |
| <b>7</b>  | <b>Hent en nøgle fra en anden</b>              | <b>9</b>  |
| <b>8</b>  | <b>Send en krypteret mail</b>                  | <b>10</b> |
| <b>9</b>  | <b>Signer en nøgle</b>                         | <b>11</b> |
| <b>10</b> | <b>Installation af FileZilla</b>               | <b>12</b> |
| <b>11</b> | <b>Putty installation - Secure Shell login</b> | <b>13</b> |
| <b>12</b> | <b>WinSCP installation - Secure Copy</b>       | <b>14</b> |
| <b>13</b> | <b>Login to Unix server</b>                    | <b>15</b> |
| <b>14</b> | <b>Get to know some Unix</b>                   | <b>16</b> |
| <b>15</b> | <b>Access the root on Unix</b>                 | <b>17</b> |
| <b>16</b> | <b>Unix boot DVD</b>                           | <b>18</b> |
| <b>17</b> | <b>Wireshark installation</b>                  | <b>20</b> |
| <b>18</b> | <b>Sniffing network packets</b>                | <b>21</b> |
| <b>19</b> | <b>Discovery using ping and traceroute</b>     | <b>22</b> |
| <b>20</b> | <b>ICMP tool - icmpush</b>                     | <b>23</b> |
| <b>21</b> | <b>Discover using DNS</b>                      | <b>24</b> |
| <b>22</b> | <b>Try the bind-version shell script</b>       | <b>25</b> |
| <b>23</b> | <b>Try the dns-timecheck Perl program</b>      | <b>26</b> |
| <b>24</b> | <b>Discover active systems ping sweep</b>      | <b>27</b> |
| <b>25</b> | <b>Execute nmap TCP and UDP port scan</b>      | <b>28</b> |
| <b>26</b> | <b>Perform nmap OS detection</b>               | <b>29</b> |
| <b>27</b> | <b>Perform nmap service scan</b>               | <b>30</b> |
| <b>28</b> | <b>Find systems with SNMP</b>                  | <b>31</b> |

|  |           |
|--|-----------|
| <b>29 Try Hydra brute force</b>          | <b>32</b> |
| <b>30 Try Cain brute force</b>           | <b>33</b> |
| <b>31 Network scripting using netcat</b> | <b>34</b> |
| <b>32 OpenSSL forbindelser</b>           | <b>35</b> |
| <b>33 OpenVAS scanning</b>               | <b>36</b> |

# Indhold

# Forord

Dette kursusmateriale er beregnet til brug på kurset *Internet Self Defence workshop*. Materialet er lavet af Henrik Lund Kramshøj, <http://www.solido.net>

Materialet skal opfattes som øvelseshæfte til kurset, og indeholder derfor ikke en fuldstændig beskrivelse af emnet. Der henvises istedet til andet materiale om emnet som nævnt i litteraturlisten.

Til workshoppen hører desuden en præsentation som udleveres.

God fornøjelse

## Oversigt

Materialet er inddelt i et antal øvelser som er beregnet til at give kursusedtagerne et indblik i hvordan Internet Self Defence praksis ser ud og opfører sig.

Formålet med workshoppen er at give deltagerne en praktisk erfaring med emnet og information til at implementere i egne miljøer.

## Forudsætninger og ordliste

Dette kursusmateriale forudsætter at deltageren har kendskab til internet og e-mail på brugerniveau. Det betyder at begreber som <http://www.solido.net>, [hkk@solido.net](mailto:hkk@solido.net) ikke bør være ukendte.

## Værktøjer

Dette materiale er udarbejdet ved hjælp af en masse værktøjer, og er beregnet på at kunne udføres i et almindeligt kursuslokale med netværksopkoblede pc'er. De praktiske øvelser benytter i vid udstrækning Open Source og kan derfor afvikles på blandt andet følgende platforme:

- UNIX - herunder Linux, OpenBSD, NetBSD, FreeBSD og Mac OS X
- Microsoft Windows 7 - primært som klientoperativsystem

Det anbefales at benytte virtualiseringsplatforme til hackerværktøjer, herunder BackTrack Linux. Der findes flere alternativer som:

- VMware Player <https://www.vmware.com/products/player/>
- VirtualBox <https://www.virtualbox.org/>
- Xen <http://www.xen.org/>

# Indholdet i øvelserne

De fleste af øvelserne har følgende indhold:

- **Opgave:** Hvad går øvelsen ud på
- **Formål:** Hvad forventes det at man lærer ved at løse opgaven
- **Forslag til fremgangsmåde:** er en hjælp til at komme igang
- **Hjælp:** er flere tips eller beskrivelser af hvordan man kan løse opgaven
- **Forslag til løsning:** en mulig løsning til opgaven
- **Diskussion:** er oplæg til diskussion efter løsning af opgaven. Der er mulighed for at sammenligne og diskutere de valgte løsninger.

## Øvelse 1

# Installation af alternativ browser

**Opgave:**

Installer en alternativ browser på din PC, eksempelvis Firefox eller Chrome

**Forslag til fremgangsmåde:**

Hent installationsprogrammet fra <http://www.mozilla.org> eller <http://www.google.com/chrome>

**Hjælp:****Forslag til løsning:**

Hent setup programmet og udfør installationen

**Diskussion:**

Vi bruger Firefox for at have en alternativ browser, som kan indstilles mere paranoidt, kan udvides med plugins, Flash blocker m.v.

Det er valgfrit hvilken browser man vælger, men en alternativ browser giver muligheder for indstilling.

Mac brugere kan derefter bruge Safari, mens Windows brugere kan fortsætte med Internet Explorer til NemID/Netbank og sites man stoler på. Jeg stoler ikke selv på NemID og har lavet en virtuel VMware med Ubuntu til dette specifikke formål.

## Øvelse 2

# Installation af Thunderbird

**Opgave:**

Installer Thunderbird mailklienten på din PC

**Forslag til fremgangsmåde:**

Hent installationsprogrammet lokalt eller fra <http://www.mozilla.org>

**Hjælp:****Forslag til løsning:**

Der findes følgende installationsfiler:

- Thunderbird Setup

**Diskussion:**

Thunderbird anbefales fremfor eksempelvis Outlook og Apple Mail grundet de gode muligheder for udvidelser, herunder GPG plugin.

På Mac OS X kan benyttes den indbyggede Mail.app med GPGMail plugin - hvis man kan leve med at den ikke altid findes til nyeste version af Mac OS X.

På andre UNIX varianter er Mutt mail reader populær og integrerer nemt til GPG.

Thunderbird giver også mulighed for nem mail filtrering med eksempelvis Sieve og Dovecot.



## Øvelse 3

# Installation af GPG GNU Privacy Guard

### **Opgave:**

Installer GNU Privacy Guard på jeres PC.

### **Forslag til fremgangsmåde:**

Hent installationsprogrammet og installer - brug pakkesystemerne hvis I bruger Linux

Mac OS X brugere kan med fordel benytte <https://www.gpgtools.org/>

### **Hjælp:**

### **Forslag til løsning:**

### **Diskussion:**

## Øvelse 4

# Installation af Enigmail plugin

**Opgave:**

Installer Enigmail plugin til Thunderbird

**Forslag til fremgangsmåde:**

Hent installationsprogrammet og installer

**Hjælp:****Forslag til løsning:**

Det nemmeste er at gå til hjemmesiden for Enigmail

<http://enigmail.mozdev.org/>

**Diskussion:**

Enigmail kræver at GNU Privacy Guard er installeret

## Øvelse 5

# Installation af Truecrypt

### Opgave:

Installer Truecrypt pakken fra <http://www.truecrypt.org/>

### Forslag til fremgangsmåde:

Hent installationsprogrammet og installer

### Hjælp:

### Forslag til løsning:

### Diskussion:

## Øvelse 6

# Lav en PGP-kompatibel nøgle

### Opgave:

Brug et valgfrit program til at lave en PGP-kompatibel nøgle

### Forslag til fremgangsmåde:

Brug Enigmail Key Manager eller PGP pakken til at generere en nøgle

### Hjælp:

Sørg for at lave din første nøgle med udløbsdato!

Sørg for at lave et revocation certificate på din første nøgle - så kan du altid trække den tilbage - selvom du glemmer kodeordet.

### Forslag til løsning:

Brug det lokale mailsetup som eksempel og lav en nøgle

### Diskussion:

Husk at hvis det skal være en rigtig nøgle skal du helst bruge din rigtige mailadresse

Jeg vil ikke anbefale at der uploades ”testnøgler” til keyservere, da man ikke kan slette sine nøgler.

## Øvelse 7

# Hent en nøgle fra en anden

### Opgave:

Find en nøgle og indlæs den i din nøglering

### Forslag til fremgangsmåde:

Brug enten en keyserver <http://pgp.mit.edu> eller en USB nøgle til at overføre en elektronisk udgave af nøglen til din PC

**husk at verificere fingerprint**

### Hjælp:

### Forslag til løsning:

### Diskussion:

## Øvelse 8

# Send en krypteret mail

### Opgave:

Send en krypteret mail

### Forslag til fremgangsmåde:

Brug PGP pakken eller Thunderbird til at sende en krypteret mail til en af de andre

### Hjælp:

### Forslag til løsning:

### Diskussion:

## Øvelse 9

# Signer en nøgle

### Opgave:

Find ud af hvordan du laver en signatur på en nøgle og returnerer den

### Forslag til fremgangsmåde:

### Hjælp:

### Forslag til løsning:

### Diskussion:

Det er en god politik KUN at signere nøgler hvor man har fået fremvist officielle papirer som kørekort og pas.

Læg yderligere mærke til at det som signaturen angiver er om den nøgle tilhører vedkommende - ikke om vedkommende er troværdig!

## Øvelse 10

# Installation af FileZilla

### Opgave:

Installer FileZilla pakken fra <http://filezilla-project.org/>

### Forslag til fremgangsmåde:

Hent installationsprogrammet og installer FileZilla, alternativt WinSCP eller et andet program der forstår SFTP

### Hjælp:

### Forslag til løsning:

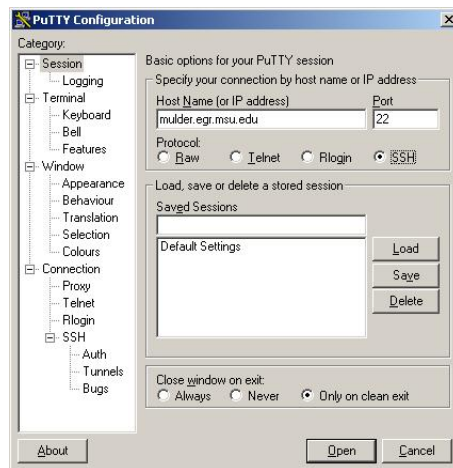
### Diskussion:

Det er vigtigt at bruge sikre protokoller når man overfører data, eksempelvis til opdatering af websites



## Øvelse 11

# Putty installation - Secure Shell login



### Objective:

Install the program Putty locally on your workstation

### Purpose:

Installing Putty will make sure you have administrative access and allow us to use Secure Shell for connecting to Unix systems and networking devices.

### Suggested method:

Download and install the program, either download from web server locally or from <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

### Hints:

Putty is a terminal emulator and replaces the telnet program in Windows. It is often the preferred way of connecting to Unix systems and is also available in network devices such as switches, routers and firewalls.

Further Putty will enable serial connections which can be used for configuring equipment through console connections. Remember to select the method when using Putty.

It is suggested to save profiles for future use, and remember to change a profile you should load the profile, make changes and **remember to go back and save the profile** before opening a connection. Otherwise the profiles changes will only be active in the current connection.

### Solution:

Do a normal installation with default settings.

If you known Putty already you can investigate the Puttygen program and research the use of public and private keys.

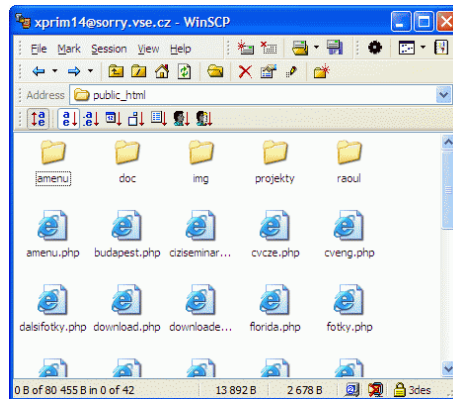
### Discussion:

The Secure Shell protocol is an internet standard for secure terminal connections and the same protocol allows file transfer and forwarding of network packets.

Note: the protocol version 2 is the one recommended

## Øvelse 12

# WinSCP installation - Secure Copy

**Objective:**

Install the program WinSCP locally on your workstation

**Purpose:**

Get required programs ready for doing exercises.

**Suggested method:**

Installing WinSCP will make sure you have access to transferring files from Unix systems and networking devices.

**Hints:**

WinSCP is very helpful allowing easy access to files using Secure Shell protocol and also when working with text files it is possible to use the built-in editor of WinSCP.

**Solution:**

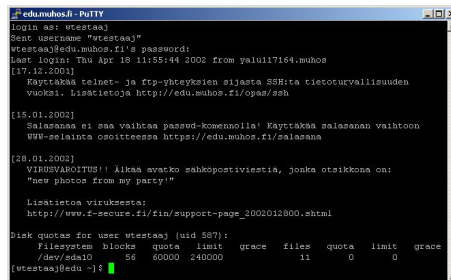
Download and install the program

**Discussion:**

WinSCP can also be used instead of FTP, why is that helpful?

## Øvelse 13

# Login to Unix server



```
edu.muhos.fi: Putty
login as: wtestaa
Sent username "wtestaa"
wtestaa@edu.muhos.fi's password:
Last login: Thu Apr 18 11:55:44 2002 from yaliu17164.muhos
[17.12.2001]
Käyttäkää telnet- ja ftp-yhteyksien sijasta SSH:tä tietoturvallisuuden
vuoksi. Lisätietoja http://edu.muhos.fi/opaar/ssh

[18.01.2002]
Salasanat ei saa vaihtaa पासवर्ड-командой! Käyttäkää salasanan vaihtoon
WWW-selainta osoitteessa https://edu.muhos.fi/salasana

[18.01.2002]
VIRUSVAROITUS!! Älkää avatko sähköpostiviestiä, jonka otsikkona on:
"new photos from my party!"

Lisätietoa viruksesta:
http://www.f-secure.fi/fi/support-page_2002012800.shtml

Disk quotas for user wtestaa (uid 597):
Filesystem blocks quota limit grace files quota limit grace
/dev/sda10 56 60000 240000 11 0 0
[wtestaa@edu ~]$
```

### Objective:

Do a remote login from your workstation to the servers provided

### Purpose:

Make sure the network is working and allow you to use the Unix system for exercises.

### Suggested method:

You will use Putty or another Secure Shell program and login to the servers provided

### Hints:

Use the Putty program or boot the Linux Live DVD and run ssh from the command line.

Using the Linux Live DVD the OpenSSH programs are already installed and available and are used with commands like this::

```
ssh username@server -p port which for the actual server is:
```

```
ssh team1@10.0.45.45 -p 22
```

NB: the server may have another IP-address due to the use of DHCP

The users defined all have the password **team**

### Solution:

Start Putty or boot using the Linux Live DVD

### Discussion:

The Linux Live CD is based on Open Source and may be copied freely.

The BackTrack security distribution contain more than 300 security programs and is being updated actively.

## Øvelse 14

# Get to know some Unix

### Objective:

Try a few Unix commands and see that help is available

Answer the following questions:

- What does the command `cal` do? What happened in September 1752?
- What does the commands `date`, `clear` and `echo` do?

### Purpose:

Learn enough Unix to be able to run simple commands from the command line

### Suggested method:

Log into the Unix system and try executing the commands

After trying the commands use the manual pages with the following commands:

`man cal`, `man date`, `man clear`, `man echo`

```
$ date
...
$ cal
...
$ cal 2009
...
$ cal 1752
...
output is not shown on purpose, try it for yourselves :-)
```

### Hints:

The manual system is always available on Unix and usually you can do searches when displaying a manual page using the operators `/` (forward search) and `?` (backward search).

### Solution:

Type `man cal` and do a search by entering `/`, the year 1752 and press enter

### Discussion:

Searching using `/` and `?` are very common on Unix

## Øvelse 15

# Access the root on Unix

### Objective:

Learn to use the `sudo` command to gain root access.

### Purpose:

Know a way to gain access as root user - to run hacker programs later

### Suggested method:

Run the command and use the manuals of the two commands `su` and `sudo` to answer the following questions:

- What is the goal of the programs?
- What are the similarities and differences?
- Can the `su` command be configured not to use a password? can `sudo`?
- What password needs to be entered when using the programs, your password or the superuser password?

### Hints:

Switch user is the old command used to gain root access - and requires the knowledge of the password for the root user or the other user your are switching to. `Su` always give complete access by switching to the user id. `Sudo` is a more modern way to control access.

### Solution:

Use the command `sudo -s` to get root access and then `exit` to exit superuser.

### Discussion:

Unix systems have traditionally used the switch user `su` - but the superuser do `sudo` is much more modern and flexible by allowing you to specify specific commands and permissions on a fine grained permission model.

`Sudo` is used almost exclusively and is considered the de facto way of gaining root on Unix systems.

An example use of `sudo` might be the restarting of a web server with apache control:

```
hlk@bigfoot:hlk$ sudo apachectl configtest
Syntax OK
hlk@bigfoot:hlk$ sudo apachectl restart
hlk@bigfoot:hlk$
```

(Note: when things succeed Unix wont say much, only if something unexpected happens there will be output)

## Øvelse 16

# Unix boot DVD



### Objective:

Boot a Live DVD on the workstation, or VMware image

### Purpose:

Learn to use Live CD's - specifically the BackTrack Live DVD

### Suggested method:

Insert the DVD and boot from it

### Hints:

There is a large number of Live CDs built on the Linux operating system specifically designed for various purposes. Some of the well known CDs are:

- Knoppix which include a lot of productivity tools, like web browser, office suite, mail programs etc.
- BackTrack which include more than 300 security tools and a premade Linux kernel with a lot of security related patches.
- Damn Vulnerable Linux which is also a security CD but the focus is on providing a learning environment for security training. Some tools help work with buffer overflows and others provide an opportunity to do reverse engineering

### Solution:

When booted use the commands shown below

### Discussion:

The Live CDs are designed to be used on most computer, but some models require more work - typically the graphic card or wireless network card can cause trouble.

If that should happen it is recommended to search on the internet, to see if others have tried using Linux on the specific brand and model of computer.

In case of the wireless card not working it is recommended to research and buy a wireless network card that is known to work.

**Note: When working with the BackTrack CD the following commands are useful:**

- `startx` will enter the graphical environment

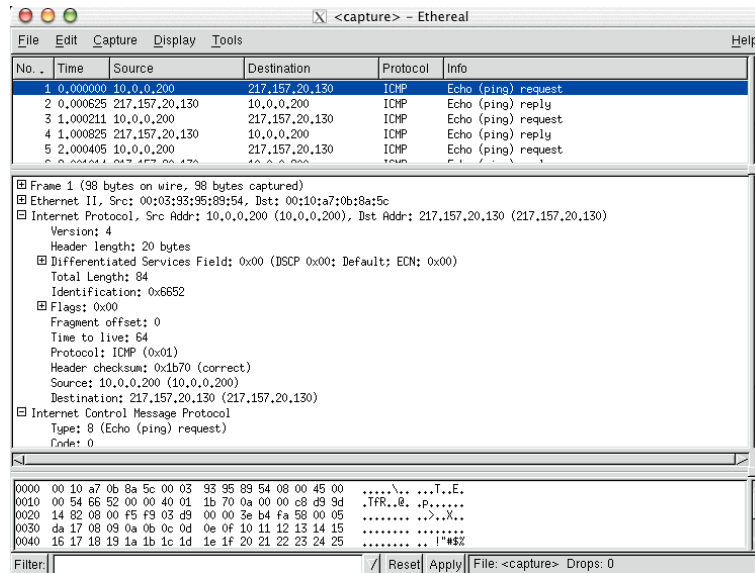
- `/etc/init.d/networking start` will try configuring the network on all interfaces with DHCP
- `dhclient eth0` start a single DHCP client using a specific network card, like eth0
- `wicd` followed by `wicd-client` will start a wireless client program to allow you to join wireless networks
- `apt-get update` and `apt-get upgrade` - upgrade when installed in hard disk
- `apt-get update` and `apt-get dist-upgrade` - upgrade with major upgrades

The individual tools on the BackTrack are described in detail on the internet and some of the tools, like Wireshark and nmap will have excellent documentation available.

**VMware settings - use bridged networking for low level tools!**

## Øvelse 17

# Wireshark installation



### Objective:

Install the program Wireshark locally on the Windows workstation

### 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 Windows Capture library to be installed, which is included in the Wireshark installation, but can none the less be downloaded from <http://www.winpcap.org/>

### 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



## Øvelse 18

# 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.

**Suggested method:**

Open Wireshark and start a capture - either from Windows or BackTrack

Then in another window execute the ping program while sniffing

**Hints:**

When running on Linux the network cards are 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?

## Øvelse 19

# Discovery using ping and traceroute

### Objective:

Learn how to use the ping and traceroute programs.

### Purpose:

Doing network discovery is an important part of doing security testing.

### Suggested method:

Use `ping` and `traceroute` testing your network connection.

Can be performed from both Windows and Unix/Linux

Remember though that traceroute is named `tracert` on Windows.

### Hints:

ICMP is the Internet Control Message Protocol which is used for reporting problems back to a source on the internet. It can also be used for diagnosing problems using ICMP ECHO request packets. ICMP is very important when doing security testing for network discovery and making sure connections are alive.

The following protocols are being used:

- Ping uses ICMP packets with request and expect responses
- Tracert on Windows uses ICMP packets
- Traceroute on Unix by default uses UDP packets, but can also use ICMP

### Solution:

Run the commands - not all are available on Windows, so perhaps use Unix:

- **traceroute** (Unix) or **tracert** (Windows)
- **traceroute -I**

**Discussion:** A lot of people just try to block any ICMP, but that will actually hurt a lot of functionality within your network.

Other trace programs exist, for example TCP traceroute programs - find them on the BackTrack!

## Øvelse 20

# ICMP tool - icmpush

**Objective:**

See a sample program that allows you to send ICMP packets without doing actual programming

**Purpose:**

Know that a lot of hacker programs exist on any level of IP

**Suggested method:**

Login to the Unix server - see the manual and use timestamp request packets

Alternative install icmpush on BackTrack using the command `apt-get`, try running icmpush and then follow on screen instructions.

**Hints:****Solution:**

Use the command `icmpush -v -tstamp 10.0.45.45` and also try echo, mask from the icmpush program

**Discussion:**

Other toolboxes for creating network packets are:

- Nemesis - which is on the BackTrack
- Scapy - which allow you to do Python programs that can send packets
- Hping - which is on the BackTrack

## Øvelse 21

# Discover using DNS

### Objective:

Try some programs for doing Domain Name System (DNS) lookups

### Purpose:

Learning to do network discovery includes looking into public information such as DNS

### Suggested method:

Try these commands:

- nslookup - available on both Unix and Windows, but not recommended anymore
- Try `nslookup -q=txt -class=CHAOS version.bind. 0`
- Try `dig @ns1.gratisdns.dk www.solido.net A`
- Try `host -a solido.net` and `host -a www.solido.net` any difference?
- The host program uses the syntax `host host server` while dig uses `dig @server host`

### Hints:

Host is available by default on OpenBSD, so use the Unix server provided

There are a lot of Graphical User Interface programs available both for Unix and Windows

### Solution:

Run the commands above, output would be like this:

```
$ host -t ns solido.net
solido.net name server ns1.gratisdns.dk.
solido.net name server ns2.gratisdns.dk.
solido.net name server ns3.gratisdns.dk.
solido.net name server ns4.gratisdns.dk.
solido.net name server ns5.gratisdns.dk.
$ host -t ns solido.net 217.157.20.131
...
```

### Discussion:

Previously it was possible to do Zone Transfers, but today most DNS system administrators do not allow that. If possible a zone transfer will reveal all names for a domain.

Make sure that you know the difference between forward and reverse lookups. Forward is from name to IP address lookup, while reverse does a lookup from IP address to name.

## Øvelse 22

# 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.

Unix shell scripting is very usefull and the book *Classic shellscripting* is recommended when doing shell scripting.

Unix also typically include scripting languages like Perl, Python, Ruby, Groovy, ...

**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
```

## Øvelse 23

# 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 these languages:

- Perl, of course :-)
- Python - like Scapy
- Ruby - like Metasploit

The script available on the system is:

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

use Net::DNS;

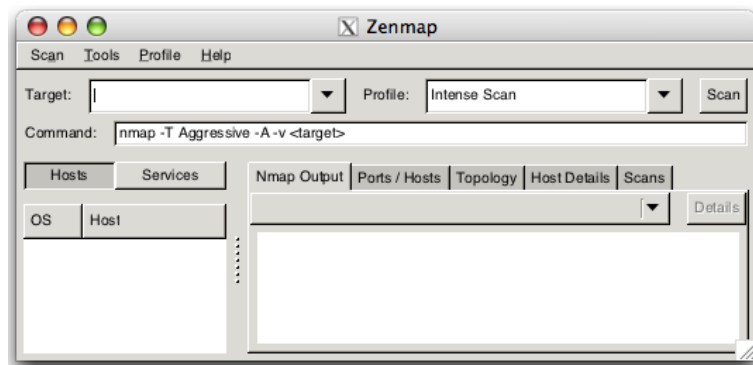
my $resolver = Net::DNS::Resolver->new;
$resolver->nameservers($ARGV[0]);

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";
    print "\n";
}
```

## Øvelse 24

# Discover active systems ping sweep



### 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.**

## Øvelse 25

# 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.



## Øvelse 26

# 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 it 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.

## Øvelse 27

# 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.

## Øvelse 28

# 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.

## Øvelse 29

# Try Hydra brute force

**Objective:**

Try a brute force program named hydra/Xhydra

**Purpose:**

Learn that some protocols allow brute forcing.

**Suggested method:**

Log into the Unix server or use the BackTrack.

Make a short list of usernames and a short list of passwords and use hydra to brute force your way into a system. Use the editor kate, using `kate users.txt` and `kate pass.txt` followed by a command similar to this:

```
$ hydra -V -t 1 -L users.txt -P pass.txt 10.0.45.2 ssh
```

**Hints:**

When learning tools create a nice environment and check that things are working before trying to hack. So with brute forcing an account, create and test it!

**Solution:**

There is an FTP server with an easy to guess administrator password.

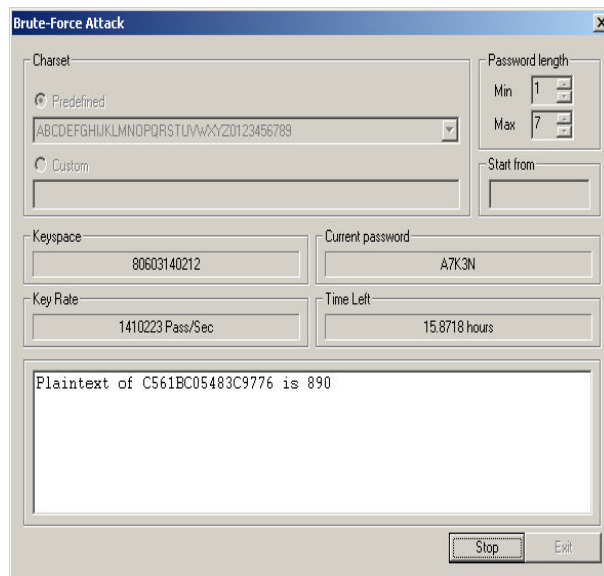
**Discussion:**

The hydra program can brute force a lot of different protocols and also allow a lot of tuning.

The hydra program does an online brute force attack, in some cases you can get access to data like password databases, or hash values that can be cracked in off-line brute force attacks.

## Øvelse 30

# Try Cain brute force



### Objective:

Try a brute force program named Cain

### Purpose:

Learn that some algorithms allow for easier brute forcing.

### Suggested method:

Download and install the Windows program Cain

Then try cracking some local accounts, access to hash is only allowed if you are administrator.

### Hints:

When learning tools create a nice environment and check that things are working before trying to hack. So with Cain use a system where you are administrator and crack local accounts.

Then later get hash values from real systems, or by doing google searches.

### Solution:

See that some algorithm can do 100.000s keys/second and others only allow 100s keys/second.

### Discussion:

Cain is built for cracking passwords in off-line brute force attacks, but also includes other features like sniffing.

## Øvelse 31

# Network scripting using netcat

**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 manual `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.

## Øvelse 32

# OpenSSL forbindelser

**Objective:** Learn how to use the OpenSSL programs to do scripting protocols wrapped in SSL/TLS

**Purpose:**

Learn that even if protocols are being wrapped in encryption you can write test programs.

**Suggested method:**

Login to the Unix server - look at the manualen `man openssl`. Note the possibility of using `openssl s_client`. Then create a textfile named `headssl.sh` using this content

```
#!/bin/sh
# get HEAD from Webserver SSL port
openssl s_client -host $1 -port $2 << EOF
HEAD / HTTP/1.0

EOF
```

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

**Hints:** Openssl programmet kan fungere som en wrapper til forbindelser til webservere og andre protokoller som benytter SSL/TLS

**Solution:**

Run the program: `./headssl.sh server 443`

**Discussion:**

Another program for SSL is `sslsan` available on the BackTrack to allow you to know the allowed algorithms on a web server running SSL/TLS.

## Øvelse 33

# OpenVAS scanning

**Objective:**

Use the OpenVAS system to do a more complex test.

**Purpose:**

See that more user friendly applications exist, but that these tools still require you to know the details.

**Suggested method:**

Create a certificate for the OpenVAS server, create a user, then start the server and client.

**Hints:**

There are a number of programs in the OpenVAS environment, but typing `openvas` and then pressing TAB twice will show you:

- `openvas-mkcert` make a certificate for the server
- `openvas-adduser` add a user
- `openvasd` start the OpenVAS server
- `OpenVAS-Client` client program that connects to the server

If you have installed BackTrack on a server make sure that you run these command as the superuser, like `sudo openvasd`

**Solution:**

Run the programs shown above in that order

**Discussion:**

Note that OpenVAS is based on the source code from Nessus. Nessus has for many years been the tool of choice for a lot of companies when doing security testing.

Unlike commercial tools which are often Windows tools that require you to bring a laptop to a specific network to allow testing this OpenVAS is based on a client-server model.

The client can be anywhere and the server only needs to be close to the network being tested.