

Introduction to Information Security - HT2020

Assignment 1 - Group 62:

Tzeni Bolena

Magd Khalil

Xavier de Verdun

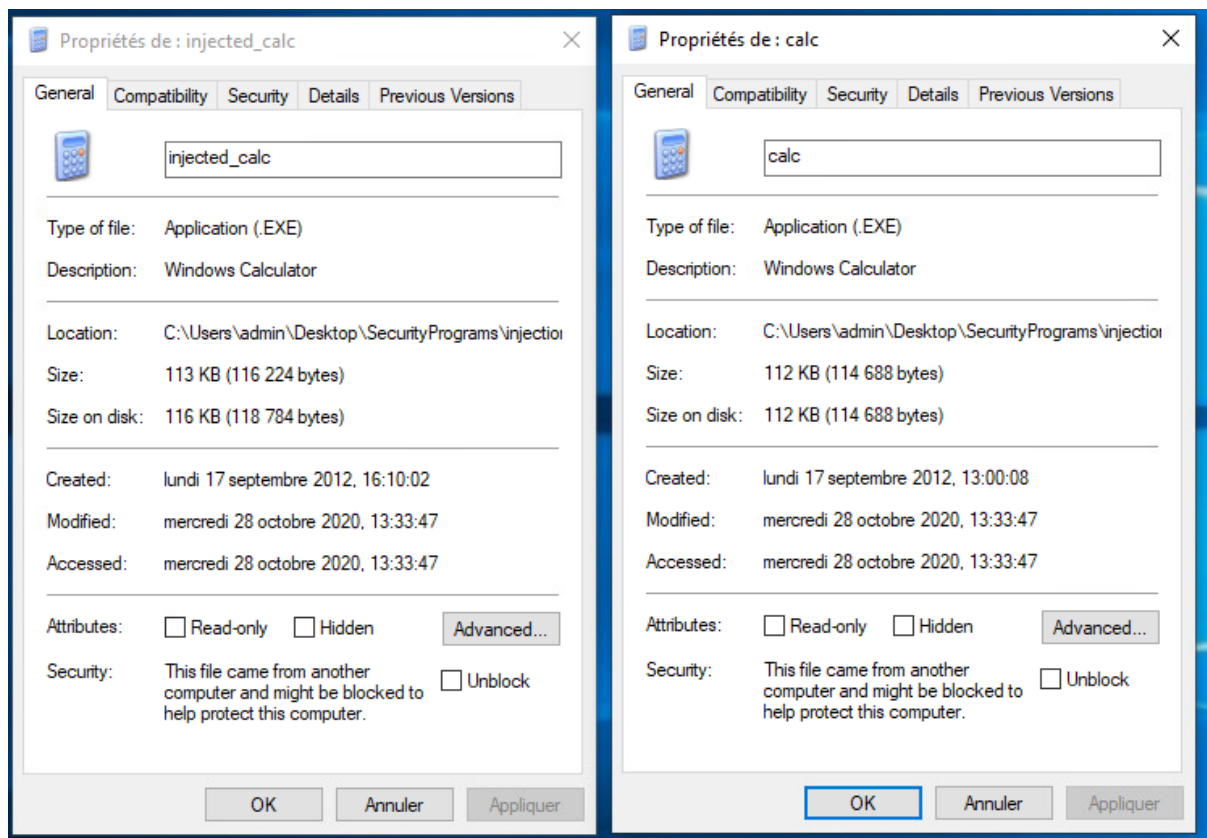


Laboratory Assignments for Windows

Exercise 1 : Code Injection

During this exercise, we analyzed two programs that look identical when they are run but in the background, they are different.

The **properties window** on Windows Microsoft 10 is a very useful interface to manage the properties of one or multiple objects selected with a GUI.



As you can see here the properties of each file “injected_calc.exe” and “calc.exe” looks quite similar except the memory size on the disk.

This gives us a clue on what and where could be the difference between those two files.

In order to go further in details on the analysis, we are going to use **fciv** command.

fciv is a command that generate a hash value for a file in MD5 or SHA-1. Then you can compare this hash to a known good one.

Here we ran fciv for each files

```
Microsoft Windows [Version 10.0.17763.1577]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\admin>cd Desktop\SecurityPrograms\fciv

C:\Users\admin\Desktop\SecurityPrograms\fciv>fciv C:\Users\admin\Desktop\SecurityPrograms\injection\injected_calc.EXE
//
// File Checksum Integrity Verifier version 2.05.
//
e1fe13a125bdcf9d2bc8651d93e9bdf8 c:\users\admin\desktop\securityprograms\injection\injected_calc.exe

C:\Users\admin\Desktop\SecurityPrograms\fciv>fciv C:\Users\admin\Desktop\SecurityPrograms\injection\calc.EXE
//
// File Checksum Integrity Verifier version 2.05.
//
829e4805b0e12b383ee09abdc9e2dc3c c:\users\admin\desktop\securityprograms\injection\calc.exe
```

The results of fciv show that the files are not identical. Therefore, one of the file might have been corrupted with an injection of code. This could potentially cause harm on a critical file.

Thus, a program may not always be what it seems! Of course, this injected program contains merely a harmless modification (as far as you know). But hidden effects may not always be so benign. What implications does this have for security?

This could be a major issue for security if the code injections are not detected at all. An injection of could be a threat for data security and applications security.

In what circumstances might an attacker (or malicious software) modify an executable as part of an attack, or utilize modified executables in an attack?

If the computer is not protected against malicious attacks such as the injection of code, then an attacker might benefit from this.

What could an executable be modified to do? How might one be exposed to such threats, and how can one protect oneself from them?

In order to take control of a computer, to scan network traffic of a computer, to extract data from a computer or corrupt it.

Use anti-malware, anti-virus software is very recommended. It is also important to have knowledge of the possible threats in your computer. Moreover, knowledge about the best practices in computer information security is very helpful to avoid any threat.

This exercise is linked to the lecture “Basics of Information Security” as we learned going through this how to protect our information.

Exercise 2 : Windows Registry

This exercise is linked to the “access control” and “authentication” part of the lectures. The Windows registry is a GUI to manage the system via the modification of critical sensitivity files. From the Windows registry, we can learn a lot and do a lot. The Windows registry is used a lot in companies to apply the same level of security and access to a group of users or an individual, remotely. Indeed, the Windows registry is one the part of the Windows system that can be hosted outside of the computer so you can prevent attacks on the registry directly on the computer.

Registry A

Part A lead us to modify the registry so the username of the last user is not shown on the login window. This is improving security as a possible attacker will need the username and the password to log in.

Registry B

Part B show us how to manage access to one user’s drives. This can be a very useful but also a very sensitive capacity. First, it can be very useful for

Registry lets you manage access to a drive for each user from the Windows Explorer. This is very useful to manage data security and data access to relevant users.

Registry C

Part C demonstrate how to manage rights for a user to run an application. However, if you can’t “run” applications with Windows key + R, you can still “open” those which are already installed, such as Google Chrome.

The Windows registry is a very powerful tool to manage information security in a way about access control to drives for users, authentication parameters and a lot of other critical systems. As it is a very powerful tool, the access to the Windows registry must be well managed so a potential attacker can’t do any harm. One way to protect windows

registry modifications directly on the computer is to move the registry to a server-hosting solution with access limited to the relevant IT specialists.

Exercise 3 : Spoofing – Bypassing the Login Screen

Exercise 3 reveals how you can easily bypass a login screen if you know where are the critical files and how to manage their properties.

sethc.exe is a program called when you click 5 times on the MAJ key.

cmd.exe is the command prompt.

In order to bypass the login screen, we did a lot of small manipulations so that an executable file “**sethc.exe**” which can be executed at the login screen, is replaced by a command prompt “**cmd.exe**” with admin rights.

This shows how the username + password authentication is now very weak of authentication. If one wants to improve the level of security for the authentication, one should use multiple authentication tools for one authentication such as username + password + phone verification, also called 2-factor authentication.

What would you be able to perform in a situation where you had access to a command shell with administrator privileges?

From the command prompt with administrator privileges, anything is possible. It's possible to run programs, modify or read or delete any kind of files.

How can the OS trust its own applications and what measures can it take to protect against such modifications?

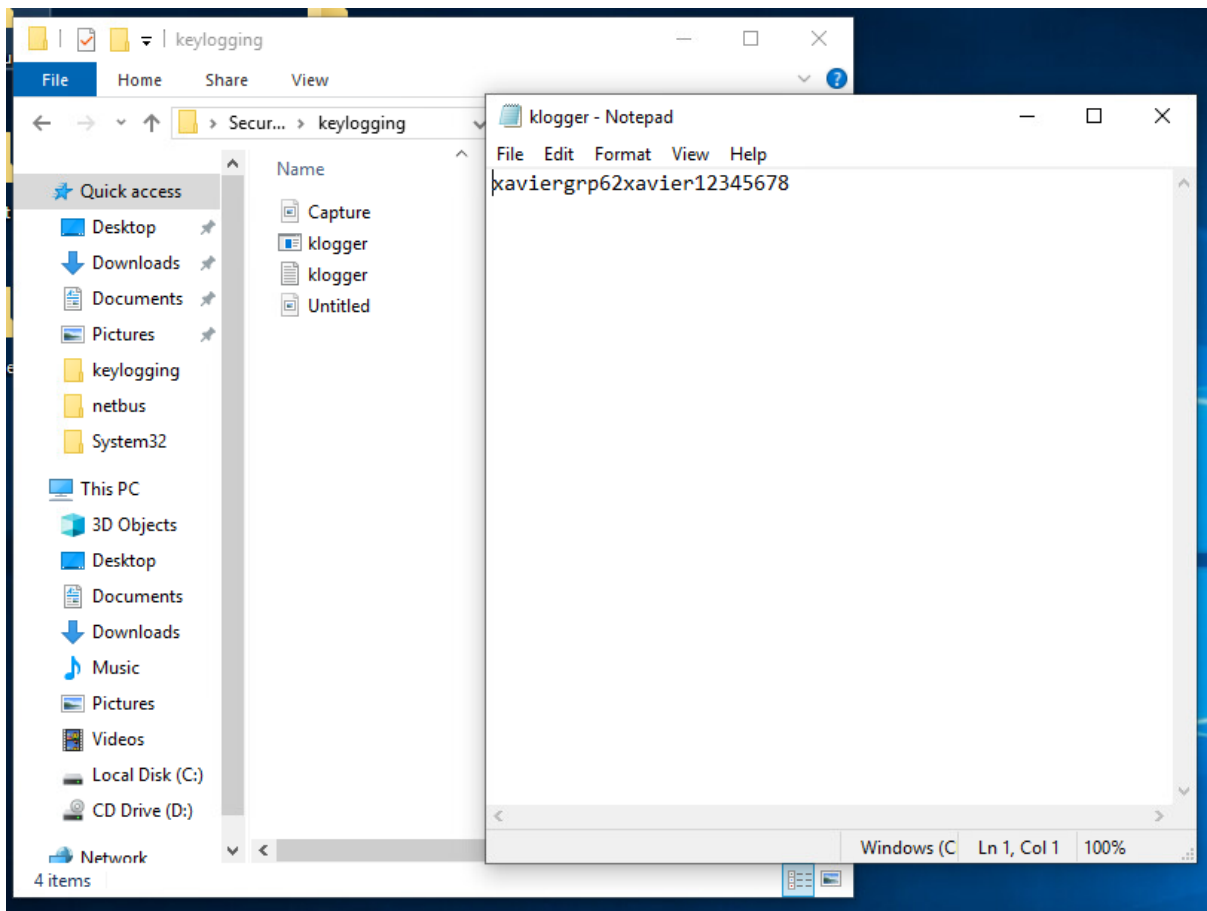
The OS should allow you only be able to access “sethc.exe” while logged in.

Exercise 4 & Exercise 5 : Monitoring keyboard strokes – Disclosing masked passwords

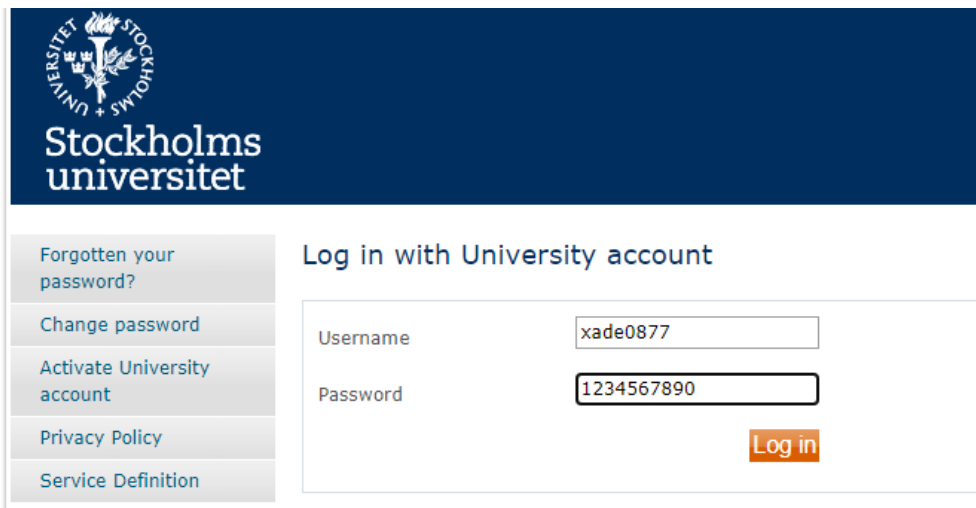
Our keyboard is the main tool to interact with the computer along with the mouse. Exercise 4 aims to use a keylogger to record every information going through the keyboard.

The keylogger is a program running in the background and saving in a file every word that you write.

During exercise 4 we are launching a keylogger and going on the use of the computer. Then the keylogger saves a file with all the keyboard inputs.



For exercise 5 the goal was to use an add on to disclose masked passwords.



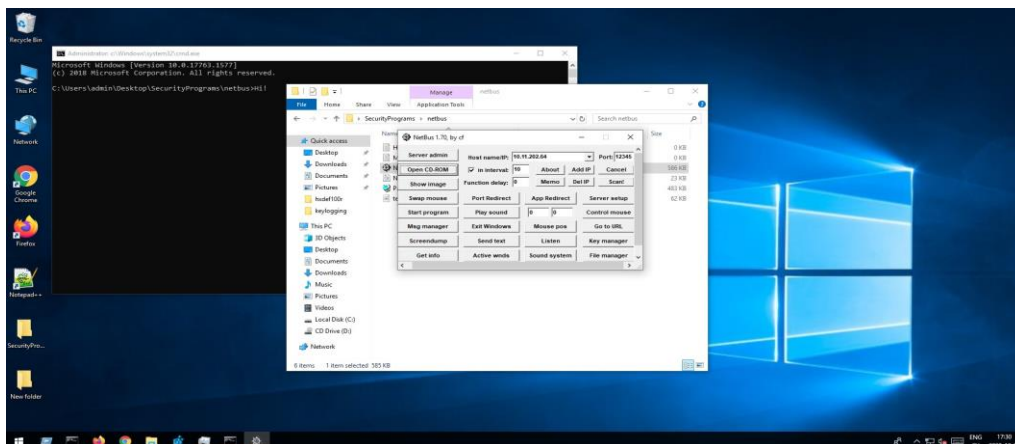
The image shows the login page for Stockholms universitet. At the top left is the university's logo and name. Below it, on the left, is a vertical menu with links: 'Forgotten your password?', 'Change password', 'Activate University account', 'Privacy Policy', and 'Service Definition'. To the right of this menu is the main login area titled 'Log in with University account'. It contains two input fields: 'Username' with the value 'xade0877' and 'Password' with the value '1234567890'. Below the password field is an orange 'Log in' button.

Here an example on SU idp.se on google Chrome.

What we learn from those exercises is that the information we pass through our keyboard to the computer, whatever the level of criticality, is very easy to access for attackers. With software directly on your computer such as the keylogger, or software linked to your web browser.

Exercise 6 : Netbus – Take control over another computer

This exercise show on another way how much an attacker can do remotely if he successfully connected two computers (the target and attacker) with Netbus software.



Netbus runs on background. That means that only your knowledge or proper anti-malware software could detect the presence of this program.

As we can see on the picture, the program can access multiple functionalities on your computer such as drives, network ports, screen dump and etc.

Laboratory Assignments for Linux

Exercise 1 : Utilising port- and vulnerability scanners

Nmap

I will start by pointing out that zenmap is based on Python 2, which has already reached its End-of-Life (EOL). It therefore is no longer being maintained and might therefore pose some security risks. I will however point out, that the community is currently working on a migration to Python 3.¹

It is interesting to see which ports are open, because open ports might be vulnerable to attacks. More specifically, the daemon (that is, the program listening to the open port) might have some vulnerabilities, which the attacker could exploit. Since some daemons even respond with their version number, the attacker might also be able to exploit a very specific vulnerability. Further information that the attacker might get access to include, but are not limited to²:

- The number of hosts and their IP and MAC addresses
- The running operating system and the version of the operating system
- The running services and the version number of the service
- The network topology (based on the response time of the hosts)

Open ports there increase the attack surface. Since it is our goal to reduce the attack surface, it is crucial to see which information an attacker might access by communicating through a specific port.

Let us start by scanning every host on the network 10.11.202.0/24, which ranges all IPs from 10.11.202.0 to 10.11.202.255). Using this, we easily obtain the following pieces of information:

1. There are a total of 27 open ports, where ports 13, 44, 1234, and 80 are open the most often
2. We obtain the protocol and the name of the service listening on those ports

If we want to run a more detailed but slower scan, we can use the following command:

```
nmap -sV -O -p- 10.11.202.2193
```

¹ Check <https://github.com/nmap/nmap/issues/1176>

² These are according to the course book

³ The meaning of the command options (flags) can be researched using the man command; -sV tells nmap to retrieve the version, -O retrieves information about the operating system and -p- will scan all ports of the given target

Where the given IP-address is the IP-address of the Metasploitable server. This command will return a table, which lists all the found port, their states, the service running on the port as well as the version of the service. The following table shows some of the entries of that table:

Port	Service	Version
21/tcp	ftp	vsftpd 2.3.4
80/tcp	http	Apache httpd 2.2.8 ((Ubuntu) DAV/2)
3306/tcp	mysql	MySQL 5.0.51a-3ubuntu5
8180/tcp	http	Apache Tomcat/Coyote JSP engine 1.1

We could now proceed and look for vulnerabilities of one of these services. For example, knowing that the Apache Server version 2.2.8 is running on Ubuntu, we know that we can cause a denial of service by sending a Range header, which has multiple overlapping ranges.⁴

The above scan is a very detailed one and returns 28 open ports for the metasploitable server. nmap (and therefore zenmap) also allows quicker scans to scan the whole network within a few minutes or even seconds. A common approach is to scan for the most common ports instead of all possible 65535 ports, such as the ports numbers 21, 22, and 80. Using this quick scan on the whole subnetwork of 10.11.202.*, we can see that there are a total of 27 open ports, ports 13, 44, 1234, and 80 being open the most often. We again retrieve the daemon and its version.

When comparing the Windows and Linux computers scan, we can see that the returned result has the same format. However, none of the Linux computer ports are open, whereas plenty of ports are open on the Windows computer.

Port	Service	Version
80	http	PMSoftware Simple Web Server 2.2
135	msrpc	Microsoft Windows RPC
139	netbios-ssn	Microsoft Windows netbios-ssn
445	microsoft-ds?	
12345	netbus	NetBus trojan 1.70
100	newacct?	

⁴ <https://www.cvedetails.com/cve/CVE-2011-3192/>

Table 1: Open ports on the Windows computer

Note that the last two ports are only open, if the trojan horse or the calculator with the injected code are running.

When scanning the firewall, we can see that all ports except for the ones shown in the following tables are closed.

Port	Service	More information
22	openSSH	PMSoftware Simple Web Server 2.2
53	Generic dns response	Microsoft Windows RPC
80	http	Supports GET, HEAD, POST and OPTIONS
443	Tcp open ssl/http nginx	
666	http	Darkstat network analyzer, supports GET and HEAD

When using the command

```
nmap -v -sA -n 10.11.202.254 -oA firewallaudit
```

we can furthermore see that the ports are unfiltered. This means that the firewall does not filter the packets and hence allows us to easily find out whether the ports are open or not. Filtered ports would include a package filter which prevents the nmap testpackages to reach the destination. This is more secure when compared to unfiltered packages because of two reasons⁵:

1. The attacker gains less information about which ports are open and which services are running
2. If the firewall simply blocks the request and does not answer at all, nmap is forced to retry to make sure that the port is not reachable because of the block and not because the network is overloaded

Overall, this exercise shows how easy it can be to retrieve information about a system and how little information we need, to get knowledge about potential vulnerabilities. System and network administrators should therefore make sure to reduce the attack surface by reducing the amount of information that is easily accessible.

⁵ <https://nmap.org/book/man-port-scanning-basics.html>

Greenbone Vulnerability Management (GVM)

When downloading the report, we receive a file, which was generated using LaTeX. It gives an overview over the vulnerabilities and rates the severity of those. For example, the windows machine is running a old version of the Simple Webserver, which fails to check the user-supplied input. According to GVM, it is therefore possible that an attack can access data and directories, which might include sensitive information. It might even be possible to execute some arbitrary code. It is also interesting to see that GVM will try to provide a solution, if it knows about one.

<code>[files]</code> <code>[Mail]</code> <code>MAPI=1</code>
Impact Successfully exploiting this issue may allow an attacker to access paths and directories that should normally not be accessible by a user. This can result in effects ranging from disclosure of confidential information to arbitrary code execution.
Solution Solution type: Mitigation Contact the vendor for a solution.

When scanning the windows computer for the first time, we can see a number of vulnerabilities, with the highest Common Vulnerability Scoring System (CVSS) being 7.8. If we however do a deep scan of the computer, GVM will reveal much more information. In this case, GVM shows that the web server will not only allow the user to traverse directories, but is also prone to buffer overflow attacks, which might give the attacker remote access to the system (CVSS of 10)

This tool is very powerful and useful for both attackers as well as system and network administrators. It gives a full report of the target system, including all open ports as well as the running programs. It then tries some common attacks and searches for known vulnerabilities. If it found some, it will report this with a reference to the national vulnerability database. This is useful for administrators, since it allows to find problems with the system rather easily. The admin can then reduce the possible attack surface by solving these vulnerabilities.

However, it is just as easy for attackers to gain and exploit this knowledge. This is similar to the problem that we face using nmap and also discussed in the course book.

When comparing the reports of the windows and the Metasploitable machine, we can see that the the Metasploitable VM has a much bigger attack surface since it has more weakpoints. Since different operating systems are build differently, they will have different security flaws aswell. The security flaws are however not only influenced by the operating system, but also by the processes running on it and the privileges of those processes. This can be observed when comparing results of the security scan for one specific port and protocol. When comparing port 80 of windows and Metasploitable, we can see that both are prone to attacks. These attacks however are different, since the daemon on the port 80 is different (Twiki for Metasploitable Simple Web Server for Windows).

We could not find any malware or trojan horses. While this was unexpected at first, it does make sense. Malware usually exploits existing vulnerabilities to infect the system, it therefore is (at least initially) not a vulnerability. Depending on the malware or trojan horse, it might however try to open ports and create more vulnerabilities. It will try to hide while doing this.

Exercise 2 : Utilising sniffer tools

Let's give a brief explanation of what sniffing is. Sniffing is a way of monitoring the internet activity of a person or a company and collecting data. Valuable information can be sniffed such as passwords, emails etc. Sniffing though can also be used in a good way by network administrators such as optimization of the network for better speed. Here we will focus on sniffing as an attack(hack) tool.

Webspy

The instructions were followed as given. The result though was not the desired one, that is that we could see on the kali browser what we are doing on the microsoft one. That might be because Kali linux might have changed the regulations. Also in the Q&A session I asked and was given the answer that some of these tools do not work properly anymore.

Lets see what i did. Following the instruction what happened is that after the command **"sudo webspy -i eth0 10.11.202.64"** and browsing in the windows browser the following were shown:

-webspy: listening on eth0

-open URL(109.105.109.251)

Which could indicate the web page that the browser was connecting to.

Different ways to get the desired performed out of webspy were used. Trying to open firefox through the command prompt could fix the issue according to a stackoverflow post, but that didn't change something either.

The main point out of this is that Webspy as a sniffer can track the internet action of a simple user and that is indicated by the fact that "open URL(109.105.109.251)" was given.

It is mainly used for monitoring emails, password etc as also pointed out in the assignment. Unfortunately it is not a very well documented tool.

Urlsnarf

"**Sudo urlsnarf -i eth0**" command was used. After the browser was opened and I started browsing on different webpages information started being logged on the console after each and one browsing step. The information included the time and date, http protocol version used. The Post/Get method used and web links. Also the browser used was shown as well as the operating system(Linux).

Logging into one website(www.ekt.gr) brought back much information. By clicking in the links brought back(depending if the website used http or https) they could be opened in the browser and either show something valid(as an image from the website) or an 404 error.

```
10.11.202.164 - - [07/Dec/2020:16:30:22 +0100] "GET http://www.ekt.gr/sites/ekt-site/files/styles/photo/public/new
s/images/EKT_EIT_NGB3.jpg?itok=VCLLZfOs HTTP/1.1" - - "http://www.ekt.gr/" "Mozilla/5.0 (X11; Linux x86_64; rv:78.
0) Gecko/20100101 Firefox/78.0"
10.11.202.164 - - [07/Dec/2020:16:30:22 +0100] "GET http://www.ekt.gr/sites/ekt-site/files/styles/photo/public/new
s/images/TechnolgyTransferOffices_EKT_Nov2020_banner.jpg?itok=jdg3L9fA HTTP/1.1" - - "http://www.ekt.gr/" "Mozill
a/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0"
10.11.202.164 - - [07/Dec/2020:16:30:22 +0100] "GET http://www.ekt.gr/sites/ekt-site/libraries/tablesorter/addons/
pager/jquery.tablesorter.pager.js?qtjxc HTTP/1.1" - - "http://www.ekt.gr/" "Mozilla/5.0 (X11; Linux x86_64; rv:78
.0) Gecko/20100101 Firefox/78.0"
```

Dsniff

When using "**ftp 10.11.202.219**" logging in and afterwards quitting the dsniff managed to get the username we entered and the password. Before that the time and both ips were shown.

We know that ftp is not an encrypted protocol and dsniff did take advantage of that by getting both username and password of the metasploitable server.

```
cs2lab@kali:~$ sudo dsniff
[sudo] password for cs2lab:
dsniff: listening on eth0

12/07/20 16:52:55 tcp 10.11.202.164.39128 → 10.11.202.219.21 (ftp)
USER msfadmin
PASS msfadmin
```

Telnet was not installed so that had to be done first(sudo apt install telnet)

When using “**telnet 10.11.202.219**” and afterward writing telnet + pressing enter we are able to execute some commands. Same as with ftp the information is not encrypted. Therefore whatever was written in the cmd was also captured and shown by the dsniff. To get a more detailed view of what can be done” help” was chosen and it gave the opportunity to see what we can do. One of those was to send streams to the server.

Ettercap

Ettercap can be seen and is a hacking tool, same as the above meaning its usage does violate security rules.

By looking at the info about ettercap it seems a very useful tool for **man in the middle attack** and the method **bridged** helps that. Both GTK2 GUI and Ncurses were checked out but GTK2 seemed more convenient so therefore it was analyzed more. When starting sniffing/attacking the option bridged sniffing is given and by research done it is suggested for when the device is connected to a LAN. In this case unified sniffing was chosen. When the process is running we can choose the targets to perform the attack on. Below we will use ettercap for the man in the middle attack assigned to us.

Man in the Middle attack

This part was done using ettercap since it was up to us to do some searching and decide what we want to do/use. A Youtube tutorial was also seen before all this([video](#)) and different articles were read to get a better understanding of the technical perspective.

It should be pointed out that to be able to get passwords and other information in this context websites on **http**(not encrypted, no cryptography techniques used) should be browsed. It seems that **https** is designed to avoid those attacks.

Lets start:

- We are using Kali Linux to run ettercap and Windows as the monitored/attacked machine.

- Who are we going to be in the middle of? In this case the router(default gateway) and machine ipv4 addresses(of Windows) were chosen.
- We open ettercap and start sniffing.
- Then we need to scan for hosts, in our case in the interface eth0.
- Afterwards we can view the host list. There are also included the router's address and the machine's Ip addresses.
- We want to be in the middle of them so we choose both of them as our target. Router->target1 & machine->target2.

```
Host 10.11.202.254 added to TARGET1
Host 10.11.202.64 added to TARGET2
```

- Then we can start an ARP Poisoning attack which will sniff remote connections.

```
ARP poisoning victims:

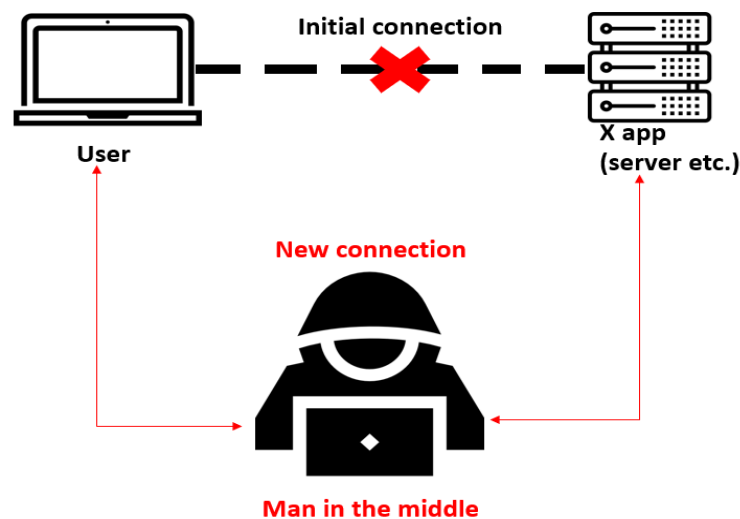
GROUP 1 : 10.11.202.254 00:50:56:80:27:DC

GROUP 2 : 10.11.202.64 00:50:56:80:CA:8D
```

- If we had access to an http website which required login we would also be able to show that the passwords are intercepted from the remote machine.

In this type of attacks so many rules are violated because of not sufficient security. Some are impersonation of another person, access to private information etc.

Sometimes understanding a concept is better when we visualize it so lets also give a visual example of what happens during all this:



Summary

All the tools mentioned above when used for hacking violate ethics and laws, those using them will be held accountable and there might be serious consequences based on the nature of the attack and the motive.

Privacy of data is also gone since the important information such as passwords are in the hands of people they do not belong to.

Also access control can be violated since the "hackers" may have gained access to software that can temper with that. Unauthorized users have access to lots of information.

Last but not least, if a system is hacked it indicates that it might need to be more secure, for example we saw that ftp and telnet are not encrypted at all in contrast to https.

Exercise 3 : Analysing network traffic

Tcpdump

Let's start by defining what Tcpdump is and then proceed to its usage. Tcpdump is a command line packet capturing utility. It helps monitor/sniff/analyze traffic over the network.

It allows the user to display TCP/IP and other packets being transmitted or received over the network to which the computer is attached. It also helps analyze and detect malicious content, package loss etc.

How do we manage to get that information though? For that let's take a look at some of the most important commands and capabilities of tcpdump which will give a better understanding on how to use it.

The base command of it is **"tcpdump -i eth0 -v"**. It gives us information over the network. "i" stands for the interface that we are using and "v" for verbose and it's used to print the info, it might be helpful if we want to give a quick look at the data. Using just this command though brings lot's of data which may be hard to analyze and understand. This is where filters come into play.

An important analysis that might be useful and we may need to do is to check all the traffic coming from a certain port to us. That port might be port 80 which is used in TCP(Transmission Control Protocol). Or otherwise we may want to check the traffic

which involves TCP and configure it accordingly in the future based on the result. The filter/command used in this case could be **“tcpdump -i eth0 -v src 10.11.202.164 and port 80”**.

```
cs2lab@kali:~$ sudo tcpdump -i eth0 -v 'src 10.11.202.164 and port 80'
[sudo] password for cs2lab:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
20:46:19.977640 IP (tos 0x0, ttl 64, id 55536, offset 0, flags [DF], proto TCP (6), length 60
)
    10.11.202.164.53797 > 10.11.202.219.http: Flags [S], cksum 0xa9c4 (incorrect -> 0x4e64),
seq 2036008513, win 64240, options [mss 1460,sackOK,TS val 38600886 ecr 0,nop,wscale 7], leng
th 0
20:46:19.977874 IP (tos 0x0, ttl 64, id 55537, offset 0, flags [DF], proto TCP (6), length 52
)
```

There are also more filters that could be applied based on what we want to get out of the capturing.

Lets also look into another capability of tcpdump. It allows us to save the captured traffic in a file(**tcpdump -w /root/Desktop/testCapture.pcap -i eth0**) and then analyze it preferably by using a GUI packet analyzing tool like wireshark which we will see later.

But we can also open the data back on the console(**tcpdump -r /root/Desktop/testCapture.pcap/testCapture.pcap**).

The output of the packages(data) depends on the protocol used. In general each packet includes:

- Time stamp
- Source and destination address
- protocol(in the image above we can see tcp(proto TCP))
- Length of package(length 52)

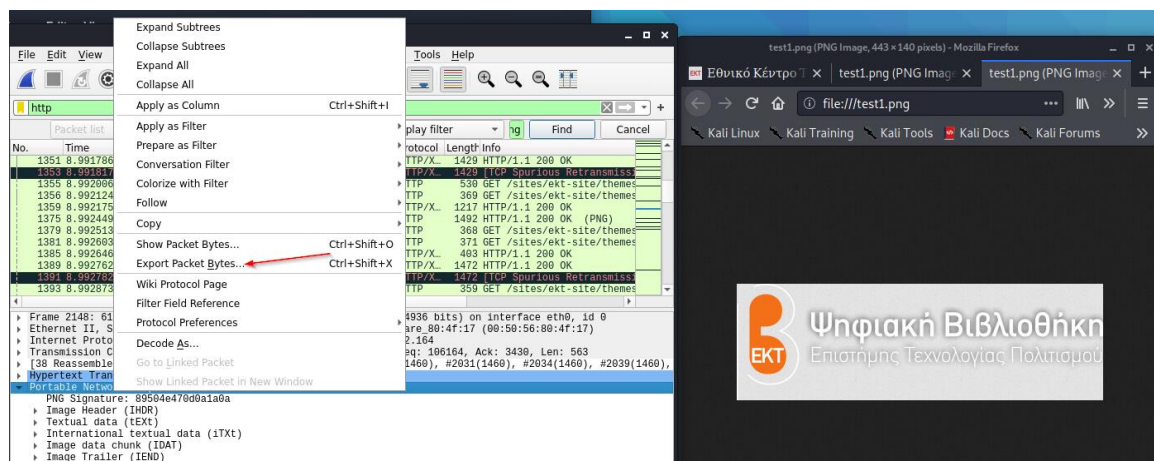
Wireshark

Wireshark is a network protocol analyzer and it captures data from a network connection. It can capture all traffic from and to your computer. As seen above tcpdump also offers this capabilities(maybe not as much analysis as wireshark). Wireshark though is a gui application, meaning we do have an interface where we can see more clearly and organize the information captured. It can be used to troubleshoot the network but it can also be used by hackers to gain usernames, passwords and more information if some of the connections over the internet are not encrypted and secure(e.g using TLS, FTP etc). Lets dive deeper by starting wireshark and capturing traffic.

We start by choosing the network interface(in this case eth0), start capture and preferably open a webpage. After loading the page we stop the capture and now it is time to observe what was captured.

We may want to analyze a specific protocol. Lets start with HTTP(Hypertext transfer protocol). We use the filter http to see only those packages. We can see information such as the frame size, the time of communication, the tcp source port and destination port and more. We said that Wireshark can be used to troubleshoot or hack. Lets see a “hack” that could be done in this case. The hacker starts capturing data while we are on the network, http protocol is used while we search on the web. That means that all data transferred is not secured since http is not encrypted and everything is shown in plain text.

We can give an example of that by showing that we can have access to the images of the website we visited.



In this case that data is not sensitive and therefore the “hacking” could have no implications. But if during the visit of the website a username and password were asked then the hacker capturing our traffic could have access to it. So it is essential to do a troubleshoot and fix issues(e.g. using https instead of http) such as that, since we do have this powerful tool in our hands. More and more analysis can be done for different protocols.

Wireshark also offers the opportunity to see how a tcp connection is established. In our capturing we can see that between two different ips in our network a 3-way handshake

connection is established which is important for future communication since it means the connection is secure and reliable. It is shown below.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.11.202.182	10.11.202.134	TCP	66	8787 -> 48525 [ACK] Seq=1 Ack=1 Win=240 Len=0 TSval=85781733 TSecr=3879112042
2	0.000265749	10.11.202.182	10.11.202.134	TCP	74	8787 -> 46087 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=85781733 TSecr=3879112042 WS=32
3	0.005575880	10.11.202.182	10.11.202.134	TCP	66	8787 -> 46087 [ACK] Seq=1 Ack=973 Win=7744 Len=0 TSval=85781734 TSecr=3879112048

Statistics are also offered to have a more gathered look over the capture. If we observe the capture already in the system(test.pcap) we can see that Ipv4 is used and 32 packets use UDP(fast transmission) while the majority(213) use TCP(lost data packets can be retransmitted).

Wireshark is such a powerful tool that someone could go on for hours by explaining its capabilities(e.g it also includes bluetooth capturing). Used for either troubleshooting or hacking it is very valuable for both parties.

SNORT

Snort is a network intrusion detection system. It runs as a packet sniffer, packet logger and an IDS(Intrusion Detection System). Lets start using it and see in more detail what it offers.

If we use one of its main commands **“sudo snort -vde -i ens160”** we can see information logged in the screen(terminal). After we terminate snort we can see in the end there are statistics we get about the entire capture(protocols used, packets I/O, memory used etc). In contrast tcpdump does not offer that in the log and there is one bonus snort has over it.

In terms of packet information lets see what snort gives.

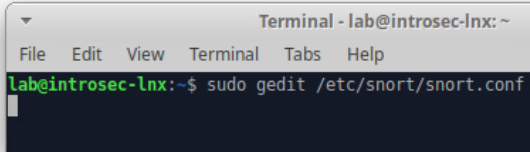
```
=====
WARNING: No preprocessors configured for policy 0.
12/06-17:59:48.950727 00:50:56:80:27:DC -> 00:50:56:80:9E:D4 type:0x800 len:0x55
172.217.21.138:443 -> 10.11.202.71:51937 TCP TTL:120 TOS:0x80 ID:46235 IpLen:20 DgmLen:71
***AP*** Seq: 0x49F69B0D Ack: 0xE5C782E6 Win: 0x109 TcpLen: 20
17 03 03 00 1A 0E 5F 83 89 C5 91 E1 46 B2 FE FE .....F...
FD 8B 7F C9 8E 74 9A 45 68 E3 4C 9F B7 7E B1 .....t.Eh.L..~.
=====
```

It has a more detailed view in comparison to tcpdump, it shows data captured too. But it can also be seen that timestamp, src ip and dst ip, the protocol used are also there as in tcpdump. Mac addresses are available, in tcmp we did not see that, but it can be done

by giving “-e” to the command. So both tools have in common that they can be used as packet sniffers and analyzers with snort giving more detailed information. Snort though is more powerful because it also works as a IDS.

Snort being an IDS means that it can detect and prevent intrusions into the systems from malicious users or software. Rules are the ones that do help the prevention and detection. We can also write our own rules. Into linux we can see them by typing “**sudo /etc/snort/snort.conf**” and by scrolling down we can see some of the rules. Not all of them are active. An example of an active rule in our case is the **icmp rules** and it helps see normal vs malicious activities on the network when it comes to the icmp protocol.

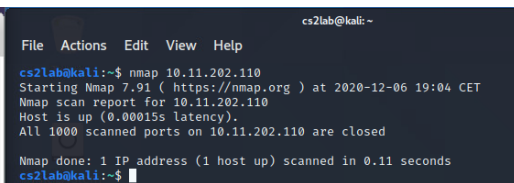
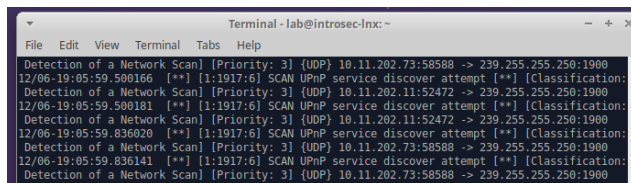
```
602 #include $RULE_PATH/file-multimedia.rules
603 #include $RULE_PATH/file-office.rules
604 #include $RULE_PATH/file-other.rules
605 #include $RULE_PATH/file-pdf.rules
606 include $RULE_PATH/finger.rules
607 include $RULE_PATH/ftp.rules
608 include $RULE_PATH/icmp-info.rules
609 include $RULE_PATH/icmp.rules
610 include $RULE_PATH/imap.rules
```



If we do an attack from another system, snort based on the rules activated will log the malicious activities by showing specific messages into the logs(the attempted request, the I, classification of detection of a network scan).

Lets show an example of logs into the terminal after using an NMap to ping. The idea came after reading the following article: [4 replies on “How to Detect NMAP Scan Using Snort”](#).

We will use Kali Linux as the device to do the NMap “attack”.



On the kali cmd we can see that nmap scanned the system and found non open ports.

Now that we have seen both Snort and Tcpdump let's conclude:

- Both are used as packet analyzers and traffic capturers and network traffic debuggers.
- Both are used via cmd(the terminal).

- Snort is more detailed when it comes to what is shown in the terminal(as mentioned above) and the information is more readable than the one in tcpdump.
- Both can be saved as a .pcap file and opened with wireshark.
- Snort does have prevention implemented into it(it is an IDS after all), that means that attacks can be stopped. On the other hand tcpdump can log the information for the network traffic and show it. After that information should be saved/logged and measures should be taken by the security team.
- A very important advantage of snort over tcpdump is that it can while live trafficking block attacks and malicious attacks/content.
- Tcpdump offers a variety of filters to be applied on the logs so we can capture only what we are interested in(e.g TCP protocol packets). On that note we should mention a big advantage of snort which is that it allows us to configure it and add our own rules.
- It is important to mention that both tools as seen do provide lots of information of what is happening over the network. Both can be analyzed and give precise information about security over the network. Using one over the other can also depend on the needs of the individual/user/business.

Summary

The above are tools that can be used both for hacking and checking the network traffic and testing the system.

In the case of non malicious use they can provide information that can lead to a reorganization of the system and a well distributed system. Also in case of testing new systems for vulnerabilities, when found those vulnerabilities can be fixed and a more secure and reliable system can be built upon the previous one.

Exercise 4: Metasploit – h4Xing made easy

This exercise shows how easy it can be to access a computer and gain full rights to the system. In this case, we were able to remote control the windows machine using the Metasploit framework with barely any prior knowledge. The Metasploit framework is a framework for penetration testing and was designed to help in creating secure systems. However, as we can see, it can also be easily misused to attack systems. An attacker can use a tool like the Greenbone Security Assistant to identify weaknesses in the target

system. We found the weakness with GVM during the deep scan of the windows machine (refer to exercise 1).

Once the attacker knows a weakness, he/she can search the Metasploit database for a way to exploit that vulnerability. Metasploit might have a program, which will allow the attacker to access the target system. Since it even comes with a lot of tools that can be used maliciously, i.e. privilege escalation, screen recording and keyloggers, it makes an attack much easier.⁶

The working hack was verified by creating subdirectories on the desktop of the hacked user. We then logged into the user account using VMRC and the user credentials and verified, that the directories were in fact created.

⁶<https://www.csoonline.com/article/3379117/what-is-metasploit-and-how-to-use-this-popular-hacking-tool.html>