Learning Outcome 1: Ethical Hacker

Contents

[Ethical Hacker 1 – Basic Hacking and Pentesting Process 2](#_Toc87535530)

[Ethical Hacker 2 – Law, Ethics and Responsible Disclosure 4](#_Toc87535531)

[Ethical Hacker 3 – Footprinting, Reconnaissance and Social Engineering 6](#_Toc87535532)

[Ethical Hacker 4 – Network scanning and enumeration 16](#_Toc87535533)

[Ethical Hacker 5 – Path traversal, File inclusion and Command injection 20](#_Toc87535534)

[Ethical Hacker 6 – SQL Injection 26](#_Toc87535535)

[Ethical Hacker 7 – XSS and CSRF 28](#_Toc87535536)

[Part 1: XSS 28](#_Toc87535537)

[Part 2: CSRF 32](#_Toc87535538)

[Ethical Hacker 8 – Network Sniffing and Spoofing – work in progress: 2 tasks left 34](#_Toc87535539)

[Ethical Hacker 9 – Password cracking 40](#_Toc87535540)

[Ethical Hacker 10 – Wireless Hacking 45](#_Toc87535541)

# Ethical Hacker 1 – Basic Hacking and Pentesting Process

* **What are the process steps of every pentest in general?**

A pentest has 5 process steps, but only the first 4 are used in general, because ethical hacking does not involve the post-exploitation (and sometimes even exploitation) part because serious and unnecessary damage can be made to the system.

These four steps are:

1. Intel gathering: during this step the pentesters gather all the information they need to proceed with the actual testing: email addresses, phone numbers, etc.
2. Footprint: this step consists of gaining more knowledge about the IT system that is going to be tested.
3. VuIn analysis: in this step some more in depth pentests are being made to look for vulnerabilities in the system.
4. Exploitation (sometimes not even this process step is used in pentesting): after a weakness has been found, if the contractor agrees, the pentester can go further and use that weakness to enter the system. This step requires maximum caution in order to prevent unwanted damage to the tested IT system.

A screenshot of a computer

Description automatically generated with medium confidence

* **What are the minimal requirements for a good pentest contract and pentest report?**

A good pentest contract is very important because of the legal issues involved. There can sometimes be a thin line between (illegal) hacking and ethical hacking. Therefore, the contract should contain the following:

1. Indemnification clause: the client allows the testers to do the pentest and becomes aware of the risks, since the testers are not liable for these risks.
2. Confidentiality agreement: it is important to keep the test and especially its results confidential. Otherwise, ill-intended people could find out about a system’s weaknesses way before the client can repair them and the system could be severely damaged.
3. Information about the tested systems: it is important for the pentesters to know precisely what they are dealing with.
4. Test origin and period: the client should know when a test is being carried out to prevent the confusion between it and an actual cyber-attack.
5. Escalation procedure: it is always indicated to think of every scenario, even the worst-case one. Both parties should know from the very beginning what would happen in case of an unforeseen event.

After the pentest is finished, the results must be written on a report.

A good report should contain the following:

1. The scope of the pentest
2. The goals of the pentest
3. The findings
4. The approach
5. Conclusions

All of this should be included in a clear and concise presentation in front of the client.

# Ethical Hacker 2 – Law, Ethics and Responsible Disclosure

* **Find a few examples of cyber-crime cases (sentences and penalties). If you are a student from abroad, find such example cases from your home country.**

My home country, Romania, is among the last countries in Europe to adopt a cyber-security set of laws. Until now I was not aware of any sentence for this crime in my country and after rigorous research, I still did not manage to find anything. However, it is quite often that public Romanian institutions get attacked because of their low security. The most common cyber-attacks against these institutions consist of obtaining passwords and leaking confidential data. Attackers from Romania and eastern Europe, especially from Russia take advantage of the poor security of our institutions’ IT systems and for now there are very little to no consequences for them.

* **Describe what you will do if you find a high risk vulnerability, unexpected, in a website or IT-infrastructure (after reading the** [**CVD policy**](https://english.ncsc.nl/publications/publications/2019/juni/01/coordinated-vulnerability-disclosure-the-guideline)**) Take into consideration if you want to make money or make the world a safer place.**

I would comply to the disclosure regulations, if applicable and would report the vulnerability found straightaway, no matter what the preconditions and the promises of the company may be. In my opinion it would not be worth the risk since there are countless other ways to make money.

* **Find two or three companies and explain the concept of responsible disclosure they have in place and compare those companies.**

1. Philips -> they put more emphasis on the reward and are very vocal when it comes to what the tester can and cannot do. Their reporting procedure is slightly more complicated in comparison to the other two companies.
2. Microsoft -> they are very specific when it comes to defining the risks and vulnerabilities found by the tester. They specify that no legal action would be taken against the tester, but no mention of a reward. They believe that the knowledge gained throughout this process is rewarding enough.
3. Carestream -> their reporting procedure is very simple and straightforward: one form the tester has to fill in. They specify that no legal action would be taken against the tester, and also a mention of a reward. They are very specific when it comes to what the tester should do and what the company should do throughout this process.

* **References:**
* Philips: https://www.philips.com/a-w/security/coordinated-vulnerability-disclosure.html
* Microsoft: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RW5Alv
* Carestream: https://www.carestream.com/en/us/services-and-support/cybersecurity-and-privacy/coordinated-vulnerability-disclosure

# Ethical Hacker 3 – Footprinting, Reconnaissance and Social Engineering

* **Explain why this is an important phase of the pentesting process.**

This is an extremely important phase of the pentesting because this is the part in which the testers gather all the information that they need to create a plan. If this phase is taken seriously and is carried out properly, then the rest of the pentest will be significantly easier and the objectives will be much clearer.

* **Make sure that you can show and explain at least 3 different techniques for searching useful information.**

1. **Google searching** -> Google can be used for hacking. By typing very specific queries, many useful and interesting things can be found and made. This is an easy way to find weaknesses on low security websites.
2. **Network information discovery** -> this technique is gathering private information from online databases and is mainly used to discover more about people or entities.
3. **Social network analysis** -> it involves manual searching on social media networks, but this process can also be facilitated by automated tools, such as Social Mapper or Maltego.

* **Find how frontpage of nu.nl looked like 10 years ago using waybackmachine.org**

Graphical user interface, website

Description automatically generated

* **Discover what URLs are hidden from search robots in robots.txt files of Pentagon and Whitehouse.**

If the domain contains a robots.txt file, the administrator of the domain can specify the URLs that should not be taken into consideration by search bots. As a consequence, the search engines will not check these URLs.

The robots.txt file of the Pentagon:

Text

Description automatically generated

The robots.txt file of the White House:

Graphical user interface, text, application, chat or text message

Description automatically generated

* **Find at least 3 people including email adress (not management) that work within a large International Company, for example using social networks like Linkedin**

1. Amanda Yaw – senior visual designer at Zappos family of companies: amandayaw@justsayyaw.com (Zappos)
2. Georgios Gousios - g.gousios@tudelft.nl (Facebook)
3. Jessie Xu - jessieunc@gmail.com (Google)

* **Use tooling for:**
* **Traceroute to determine path to fontys.nl, fhict.nl**

Traceroute to fontys.nl: using the command tracert followed by a URL you can display possible routes and measure the transit delays of packets across an IP.

Text

Description automatically generated

Traceroute to fhict.nl: applying the same principles to a different website this time:

Text

Description automatically generated

* **determine which DNS and email servers are used by fontys and fhict:**

DNS and email servers used by fontys: in order to find out this information, the nslookup instruction is required. After that, I wrote fontys.nl after the generated text.

Text

Description automatically generated

DNS and email servers used by fhict: I have done the same thing, but typed fhict.nl at the end instead:

Text

Description automatically generated

* **determine which ip-adressess are used by fhict by whois utility and non whois tool**

IP addresses used by fhict:

**Whois utility:**

I have searched whois on google to find the website whois.com. I have searched for fontys and the result is this:

Table

Description automatically generated

Table

Description automatically generated with low confidence

**Non whois utility:**

IP adress of fontys.nl:

Text

Description automatically generated

* **run the harvester utility for domain fontys.nl**

Running the Harvester utility by typing the command “theHarvester” in the kali machine terminal:

-d represents the domain, -l represents the limit of the number of results, and -b represents the search engine:

Text

Description automatically generated

Text

Description automatically generated

References:

* <https://www.gohacking.com/use-google-for-hacking/>
* <https://web.archive.org/web/20110225155609/http://www.nu.nl/>
* <https://www.whois.com/whois/fontys.nl>
* <https://www.defense.gov/robots.txt>
* <https://www.whitehouse.gov/robots.txt>

# Ethical Hacker 4 – Network scanning and enumeration

Credentials:

A screenshot of a computer

Description automatically generated

* **Read, show, and explain about the different type of portscanning techniques.**

The terminal has found 37 out of the 256 possible hosts in the 192.168.167/24 network running by using the -sP scanning. It also works for finding my own hosts:

Graphical user interface, text

Description automatically generated

Example of a TCP SYN scan:

Text

Description automatically generated

example of a TCP connect scan:

Text

Description automatically generated

The -sT scan type is for the TCP connection scanning. The server receives SYN from the client to find out if the server is there. After that, the server sends a response containing SYN/ACK to notify the client of its presence. Finally, the client sends ACK to notify that they are ready to establish a communication. Nmap can find out if a port is open or closed by using this 3-way handshake. In the given example, the ports 80 and 443 have been chosen because they are used to deliver http services.

example of a UDP scan:

Text

Description automatically generated

The -sU scan type is for the UDP connection scanning. It sends UDP packet for the targeted port. The port is close if an ICMP (Internet Control Message Protocol) “port unreachable” is returned. Otherwise, the service responds with a UDP packet.

* **Read, show, and explain about** **service scans for enumeration**.

Example of a version detection scan:

Graphical user interface, text

Description automatically generated

The detection command searches through open ports to find useful information about what is currently running and how. This example shows the service and version run on port 80.

* **Read, show, and explain how you can detect the OS of a system.**

**Example of an operating system detection scan:**

Text

Description automatically generated

The OS detection scan is run using the -O flag. Nmap sends multiple TCP and UDP packets to the remote host and examines the responses thoroughly. After multiple tests have been performed, nmap searches through the contents of its database to look for a match.

# Ethical Hacker 5 – Path traversal, File inclusion and Command injection

* **What is path traversal and how does it work?**

Path traversal is an attack through which the content of files outside the root directory is shown. The most common method is using the “../” sequence. Path traversal is not as common as SQL injection, for example, but is still a threat for the web applications.

* **What is remote file inclusion and how does it work?**

Remote file inclusion is a method used by attackers to get the web application to include a remote file. Malicious code can also be entered for the web application to execute it if there is no security in place for such input. Below there are some examples of file inclusion attacks:

Modifying the URL (page=*file4.php*) to find a hidden file:



Modifying the URL (using the “../” sequence repeatedly) to get the user account:

Graphical user interface, text, application, Word

Description automatically generated

Modifying the URL to get the OS type:

Text

Description automatically generated

Modifying the URL to find the TCP connections:

Graphical user interface, application

Description automatically generated

Modifying the URL to find the UDP connections:

Graphical user interface, text

Description automatically generated

Modifying the URL to include the Google search page to the initial web page:

Graphical user interface, text, application, chat or text message

Description automatically generated

* **What is command injection and how does it work?**

Command injection is a form of attack in which the commands are transmitted by exploiting one of the application’s vulnerabilities, most commonly through user input, when the input validation is not sufficient or non-existent. Below, there are a few examples of command injection attacks:

Browsing the directory using the *ls*

Text

Description automatically generated

Using the *cat* command and index.php to create a new form:

Text

Description automatically generated

Using command injection to find the name of the user, the host name, and the IP address:

Text

Description automatically generated

* **How are path traversal, remote file inclusion and command injection related?**

All of these methods can be used on web applications, via the user input. They mainly make use of the lack of input validation, which is relatively easy to solve but it still is an important security issue among many web applications.

* **Resources:**
* [**https://www.acunetix.com/blog/articles/path-traversal/**](https://www.acunetix.com/blog/articles/path-traversal/)
* [**https://www.acunetix.com/blog/articles/remote-file-inclusion-rfi/**](https://www.acunetix.com/blog/articles/remote-file-inclusion-rfi/)
* [**https://www.imperva.com/learn/application-security/command-injection/**](https://www.imperva.com/learn/application-security/command-injection/)

# Ethical Hacker 6 – SQL Injection

* **A lot of interesting MySQL challenges are available where you can to 'create' programming code containing SQL statements.**
  + **DVWA - SQL injection & Blind SQLi (style 1) -> Objective: There are 5 users in the database, with id's from 1 to 5. Your mission... to steal their passwords (hashes) via SQLi.**

Performing an injection attack to get all 5 users: the data in a form is between ‘’ these characters. If one of them is typed in the form, it counts as the end of the message. What goes after is considered actual code and it will be executed. If an information that is always true, such as 1=1, all information will be displayed:

Table

Description automatically generated with medium confidence

Figuring out the number of columns: using the same technique, typing an SQL command after the ‘ character, it will be executed.

Shape

Description automatically generated with medium confidence

In this instance, the table has 2 columns.



Typing the command below inside the textbox will generate this information:

**' UNION SELECT user, password FROM users#**

A picture containing text

Description automatically generated

**Resources:**

* <https://portswigger.net/web-security/sql-injections>

# Ethical Hacker 7 – XSS and CSRF

## Part 1: XSS

* **What is cross-site scripting and how does it work?**

Cross site scripting is a security vulnerability found in some web applications. Cross site scripting attacks are a way for the attackers to inject client-side scripts into the web pages. Such a cross site scripting vulnerability can be used by attackers for a variety of purposes, such as bypassing the access controls. However, given the large amount of possibilities, there is a great variety in risk of these attacks depending of the sensitivity of the data handled.

Example of cross site scripting: first doing what I am supposed to do with this form to see the result:

Reflected Cross site scripting:

Graphical user interface, text, application

Description automatically generated

Typing <script>alert("The script in the user input works!")</script> in the textbox:

Graphical user interface, text

Description automatically generated

Typing <script>alert(document.cookie)</script> in the textbox:

Graphical user interface, text, website

Description automatically generated

These above are examples of injection via the input form.

Stored XSS: initial completion according to the purpose of the form:

Graphical user interface, text, application, email

Description automatically generated

Injecting script code in the message field:

Graphical user interface, text, application, email

Description automatically generated

Typing the text above will lead to this:

Rectangle

Description automatically generated with medium confidence

* **Explain the different types of XSS attacks: persistent versus non-persistent.**

The screenshots above are made for stored and reflected cross site scripting. As a matter of fact, persistent XSS is another name for stored XSS. This a more devastating sort of XSS vulnerability. It occurs when the data provided by the attacker is saved by the server and becomes a regular part of the web page, for every user to see. The screenshot above, showing the message “It works!” is displayed every time I am opening the respective DVWA page. It has been saved and it is an integral part of that page.

Non-persistent XSS or reflected XSS is the most common type of web vulnerability. It can be noticed when the data provided by the user, usually in a form, is taken and used by the server and displays the results immediately without properly sanitizing the content.

* **Give some other examples of XSS from Escape Alf.**

This example is very similar to an SQL injection attack. In the input field, the first characters to be typed in represent the characters in the actual code that mark the end of the input text. What is written after these characters is considered actual code and will be treated accordingly.

Graphical user interface, text, application, email

Description automatically generated

* **XSS exploitation possibilities conclusion: what could an attacker do if XSS is possible?**

There is a wide range of possibilities when it comes to XSS attacks. They could use persistent XSS to permanently modify the contents of a web page, at least until the code is cleaned up. The attacker could also use different XSS types in order to steal cookies, harvest credentials, force downloads, record audio or take pictures, and many more.

* **How to prevent XSS attacks?**

XSS filters need to be created in order to sanitize all the data the website receives. The main points of data entry that require sanitizing are the URL, the cookie data, the headers data and the database data, if not properly validated on user input.

## Part 2: CSRF

* **What is CSRF and how does it work?**

Cross site request forgery is a type of a malicious exploit of a web page in which unauthorized commands are submitted by a user that the web application trusts. It can be transmitted through hidden forms, specially crafted image tags, or javascript XMLhttp requests. This makes it possible to work without the user acknowledging it. In a CSRF attack the user is tricked by an attacker into submitting a web request they did not intend. This may cause actions to be made on the website that can include even client data leakage, change of session state or manipulation of an end user’s account.

The input data can be seen in the URL. This is a severe vulnerability, since the information is taken from that URL, so whatever is written there will be registered as valid.

Graphical user interface, text, application, email

Description automatically generated

If I modify the URL, I can change the password. Sensitive information such as the password in this case can be easily changed without authentication being required.

Graphical user interface, text, website

Description automatically generated

If I log in using the new credentials in the URL, it will work. The old password is no longer available. I have changed the user password using a CSRF attack.

A picture containing graphical user interface

Description automatically generated

* **Explain the different way of cross site request forgery attacks.**

There are several ways of CSRF attacks. The most common ones are modifying GET or POST statements. Before the attack, the attacker studies the website so the attack can blend in the code without causing any suspicions. Usually, for a GET statement, a <a> tag is modified in the HTML code, by embedding a bad request in a hyperlink. For a POST statement, a CSRF attack can be delivered by modifying code inside a <form> tag.

* **Resources:**
* [**https://resources.infosecinstitute.com/topic/how-to-prevent-cross-site-scripting-attacks/**](https://resources.infosecinstitute.com/topic/how-to-prevent-cross-site-scripting-attacks/)
* [**https://en.wikipedia.org/wiki/Cross-site\_scripting#Related\_vulnerabilities**](https://en.wikipedia.org/wiki/Cross-site_scripting#Related_vulnerabilities)
* [**https://en.wikipedia.org/wiki/Cross-site\_request\_forgery**](https://en.wikipedia.org/wiki/Cross-site_request_forgery)
* [**https://www.imperva.com/learn/application-security/csrf-cross-site-request-forgery/**](https://www.imperva.com/learn/application-security/csrf-cross-site-request-forgery/)

# Ethical Hacker 8 – Network Sniffing and Spoofing – work in progress: 2 tasks left

* **What can and cannot be seen when sniffing traffic with Wireshark?**

On the capturing screen, the user can see the time of capture, the number of each package, the source (the IP address of the device where the traffic comes from), the destination (the IP address that the package is trying to reach), the protocol, the length of the package, and additional information

Graphical user interface

Description automatically generated

* **Try to intercept a plain text password (HTTP or FTP) by capturing a login-name and password:**
* The first step is to enter on a non-secured website with a login form. In this example, I am using one of my older websites made for a semester 2 assignment.

Graphical user interface, application, website

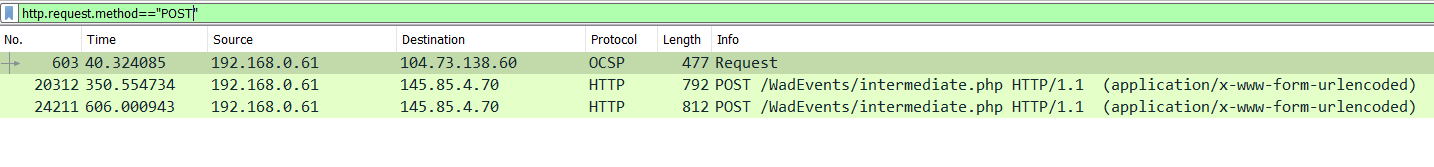
Description automatically generated

This is followed by typing a random username and password. In this case, the username is “something” and the password is “somethingelse”.

Graphical user interface, text, application

Description automatically generated

After filtering the traffic using the instruction on the green background, the name and file in question of the website is present in 2 of the 3 entries.



Now that the correct package has been found, the following instructions will lead to more information about it:

Graphical user interface, text, application, email

Description automatically generated

The username and password can be seen in the Follow HTTP Stream page, and they have been highlighted by me:

Graphical user interface, text, application

Description automatically generated

* **Demo and explain ARP spoofing a private VLan in the seclab with arpspoof.**

First, I will find out what my IP address is, using *ipconfig* on my windows command prompt:

A picture containing text

Description automatically generated

A picture containing text

Description automatically generated

I will open 2 terminals in the kali machine because I have to initialize the ARP spoof twice. In the following command, i stands for interface, in this case wlan0, t is for target, which is the windows IP address, followed by the default gateway.

Text

Description automatically generated

To be continued…

* **Explain what can be done to protect against ARP spoofing.**

There are several ways you can prevent ARP spoofing attacks. One of them is to invest in detection software that will allow you to spot potential threats. Alternatively, you could set yourself up with packet inspection software, that will check all the data on the network perimeter. This type of filtering can usually be performed by a firewall. Another option would be to create a static ARP address for every device on the network. Static ARP addresses cannot be manipulated by ARP reply packets. Although useful, this will only mitigate the most basic of threats.

* **Resources:**
* [**https://proprivacy.com/blog/ARP-spoofing**](https://proprivacy.com/blog/ARP-spoofing)
* [**https://www.youtube.com/watch?v=lb1Dw0elw0Q**](https://www.youtube.com/watch?v=lb1Dw0elw0Q)
* [**https://www.youtube.com/watch?v=FcwrrAT1H7o**](https://www.youtube.com/watch?v=FcwrrAT1H7o)
* [**https://www.guru99.com/wireshark-passwords-sniffer.html**](https://www.guru99.com/wireshark-passwords-sniffer.html)
* [**https://www.golinuxcloud.com/man-in-the-middle-attack-arp-spoofing/**](https://www.golinuxcloud.com/man-in-the-middle-attack-arp-spoofing/)
* [**https://www.youtube.com/watch?v=8SIP36Fym7U**](https://www.youtube.com/watch?v=8SIP36Fym7U)

# Ethical Hacker 9 – Password cracking

* **What is password cracking?**

Password cracking is the process of recovering passwordsfrom data that has been stored in or transmitted by a computer system in scrambled form. A common approach (brute-force attack) is to repeatedly try guesses for the password and to check them against an available cryptographic hash of the password or a list of passwords.

* **How does password cracking work?**

I began by logging in on DVWA and went to the **Brute Force** page. I typed my username and a wrong password on purpose, then pressed the submit button. After this, I opened Burp Suite and went to the proxy tab.

I looked at the request and response for this specific request and sent this line to the intruder:

A screenshot of a computer

Description automatically generated

Adding a payload marker on the password field:

Graphical user interface, text, application, email

Description automatically generated

Adding the most common 100 passwords on the simple list:

Graphical user interface, text, application, email

Description automatically generated

I went back to DVWA and intentionally typed a wrong password so I can copy the message received.

A screenshot of a computer

Description automatically generated

I have pasted the message in the grep section so that the incorrect passwords, which will generate this message, will be flagged:Graphical user interface, text, application

Description automatically generated

I then went to the attack section, with this part automatically filled in, and began the attack.

Graphical user interface

Description automatically generated with medium confidence

The attack tries all the passwords, and if the **Username and/or password incorrect** is displayed, the box will be ticked. The only unchecked password in this case will be the correct one.

Table

Description automatically generated

There are many alternatives to this. For instance, the message generated by typing the correct password can be copied and pasted, and then only the correct password will be checked.

* **Give examples of famous password cracking attacks.**

On June 5th, 2012, LinkedIn has been hacked and the passwords for nearly 6.5 million user accounts have been stolen by cyber criminals. As a result, a United States grand jury indicted Nikulin and three unnamed co-conspirators on charges of aggravated identity theft and computer intrusion.

In 2013, Tumblr has been attacked by hackers. 65,469,298 unique emails and passwords were leaked from Tumblr. The data breach's legitimacy is confirmed by computer security researcher Troy Hunt.

* **How to make password cracking harder for an attacker?**

The easiest way to make password cracking harder for an attacker is to use a long password which contains lowercase and uppercase letters, numbers, and special characters. This makes the password cracking process extremely lengthy and difficult.

* **Resources**
* [**https://en.wikipedia.org/wiki/Password\_cracking**](https://en.wikipedia.org/wiki/Password_cracking)
* [**https://en.wikipedia.org/wiki/List\_of\_security\_hacking\_incidents**](https://en.wikipedia.org/wiki/List_of_security_hacking_incidents)

# Ethical Hacker 10 – Wireless Hacking

* **Crack WPA (2) with airmon-ng and aircrack-ng from Kali Linux**.

I have downloaded a new kali virtual machine using this link: [https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/](https://eur01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fwww.offensive-security.com%2Fkali-linux-vm-vmware-virtualbox-image-download%2F&data=04%7C01%7Cf.pana%40student.fontys.nl%7C617e498c0bf14179a54708d98e354c4c%7Cc66b6765b7944a2b84ed845b341c086a%7C0%7C0%7C637697182641574247%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000&sdata=2wyc%2BX%2F5r9PJhmoTUGr%2FU9DuisA1r8Vm5Go9LWS8Ecs%3D&reserved=0)

I have installed it and entered it. Afterwards, I have used s WiFi USB-stick Netgear WNA1100, which I have borrowed from the ISSD.

Afterwards, I have started the wireless interface using airmon-ng:

Graphical user interface, application

Description automatically generated

Then, I am typing iwconfig to make sure there are no other athX interfaces.

Text

Description automatically generated with medium confidence

Using airmon-ng again, this time in order to start the wireless card on channel 9 in monitor mode:

Text

Description automatically generated

Using airodump-ng to discover the information I need from the wi fi network that I am about to attack: megabeast1:

A picture containing text

Description automatically generated

Copying the authentication handshake and pasting it next to the capture instruction from airodump-ng:

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

The handshake has happened after opening a separate terminal and using the aireplay-ng instruction along with the authentication handshake. The initial terminal is notified and displays the WPA handshake on the botton left of the screenshot:

Graphical user interface, text

Description automatically generated

After obtaining the needed information and creating the handshake, I can now stop wlan0mon and move on to the next step: unzipping the rockyou.txt file:

A screenshot of a computer

Description automatically generated with medium confidence

Using the aircrack-ng instruction to start the password cracking process:

Text

Description automatically generatedText

Description automatically generated

The process has started, the program tries different elements of the rockyou file until it finds the one that matches the actual password. In this case, the correct password is Zanzibar, and it has been cracked after 2 minutes and 57 seconds.

A screenshot of a computer

Description automatically generated with medium confidence