Gent Thaqi

Creating Labs for Ethical Hacking Course Based on WiFi Pineapple and USB Rubber Ducky

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Bachelor's Thesis

April 2024

Abstract

Author:	Gent Thaqi
Title:	Creating Labs for Ethical Hacking Course Based on WiFi Pineapple and USB Rubber Ducky
Number of Pages:	36 pages + 2 appendices
Date:	April 2024
Degree:	Bachelor of Engineering
Degree Programme:	Information Technology
Professional Major:	Smart Devices and IoT
Supervisors:	Marko Uusitalo, Senior Lecturer

As technology evolves and cyberattacks increase, cybersecurity education is more crucial than ever. Ethical hacking, or penetration testing, plays a vital role in safeguarding networks by proactively identifying vulnerabilities. This project focuses on developing hands-on labs within an Ethical Hacking course, utilizing tools like the WiFi Pineapple and USB Rubber Ducky to vividly illustrate real-world cybersecurity risks and countermeasures.

The primary goal of this work is to design labs that are both informative and engaging, moving beyond theoretical concepts to provide direct experience with industry-standard tools. Students will gain a deeper understanding of public WiFi vulnerabilities and common cybersecurity threats, equipping them for future cybersecurity careers. This thesis will detail the hardware, software, and theoretical framework that guide the lab design.

The completed labs demonstrate the power of these tools while highlighting the vulnerabilities they exploit. Importantly, they will also cultivate a strong sense of ethical responsibility among students. The aim is to foster future cybersecurity professionals who possess both technical prowess and an unwavering commitment to ethical conduct.

In conclusion the project resulted in the development of two labs for ethical hacking students which gradually increase in complexity, as well as recommendations for future iterations of the labs. The lab documents can be found in the appendix of this document.

Keywords: Ethical hacking, WiFi Pineapple, USB Rubber Duck, Cybersecurity Education, Lab Development and Design.

The originality of this thesis has been checked using Turnitin Originality Check service.

Contents

List of Abbreviations

AP:	Access Point
BSSID:	Basic Service Set Identifier
DPAPI:	Data Protection API
EU:	European Union
GDPR:	General Data Protection Regulation
HID:	Human Interface Device
HIPAA:	Health Insurance Portability and Accountability Act
HTTPS:	Hypertext Transfer Protocol Secure
IP:	Internet Protocol
IT:	Information Technology
LED:	Light Emitting Diode
OS:	Operating System
PCI DSS:	Payment Card Industry Data Security Standard
SID:	Security Identifier
STEM:	Science, Technology, Engineering, and Mathematics
VPN:	Virtual Private Network
WAN:	Wide Area Network

DHCP: Dynamic Host Configuration Protocol

SoC: System on a Chip

MitM: Man-in-the-middle

DNS: Dynamic Name System

1 Introduction

As technology makes leaps forward, so does the creativity of malicious cyber actors. As such, the importance of cybersecurity has only gotten more significant in this digital age. A crucial component of the field, ethical hacking, strives to enhance network defences and facilitates proactive identification of vulnerabilities, not to let malicious actors exploit them.

Also known as penetration testing, ethical hacking is not only a tool for cybersecurity professionals, but also an educational discipline that aims to prepare the next generation of security professionals to think like hackers and act as defenders. Moreover, ethical hacking courses are essential for a well-rounded Information Technology career path, even for students who are not necessarily aiming to work in the field of information security; regardless of the subfield that an IT professional chooses to pursue. Being able to recognize vulnerabilities and prevent security breaches in a network environment remains an indispensable skill in the job market.

Therefore, the significance of practical labs cannot be overstated, as they provide students with the opportunity to have hands-on experience within controlled environments, in conjunction with the theoretical knowledge provided by cybersecurity education.

The purpose of this thesis is to explore and articulate the process of creating practical labs for an Ethical Hacking course, specifically focused on the use of the following tools: WiFi Pineapple, USB Rubber Ducky, and external WiFi cards capable of a monitor mode. The aim is to illustrate the simplicity of hacking in real-world scenarios, thereby enhancing students' understanding of the vulnerabilities associated with public WiFi, weak passwords and more, while also fostering a deep awareness of the importance of cybersecurity in the digital age.

This thesis aims to enhance cybersecurity education by developing lab exercises that utilize the WiFi Pineapple and USB Rubber Ducky devices. The primary goals are to create the lab exercises themselves, pinpoint the skills these labs will impart, and teach how students can protect themselves against similar attacks.

The thesis is structured as follows: following the introduction, the methodology and materials section will lay out the hardware and software components involved in the labs, alongside the research methods that were utilized to guide the design of the labs. Thereafter, a theoretical framework will be established, examining the fundamentals of ethical hacking as a discipline, the functionalities of WiFi Pineapple and Rubber Ducky and the attack vectors that the two tools provide. Once the theoretical framework is laid out, the thesis will discuss the design process for each of the labs in detail.

2 Methodology and Materials

This chapter outlines the essential materials and the methodological approach used in the design and the theoretical evaluation of the ethical hacking labs proposed within this thesis. It outlines the specific hardware and software components underpinning the creation of the lab environments, the methodology behind the study, as well as the constraints it presented.

The design and implementation of the ethical hacking labs within this thesis rely on several key hardware and software components. The core hardware includes a WiFi Pineapple (Mark VII and Enterprise models), a USB Rubber Ducky, an external Wi-Fi adapter that supports monitor mode, a target machine (running on Windows 11) and a host machine (running on a Linux distribution).

In terms of software, suitable tools for configuring and utilizing the WiFi Pineapple and USB Rubber Ducky will be needed, such as Python3, the Metasploit framework, Payload Studio, GitHub, WiFi Pineapple Stager, and Windows batch files .bat amongst others.

In addition to the hardware and software that is directly related to the labs, for the purposes of the development of the theoretical framework, various literature sources were utilized.

Due to the time limitations and scope of this project, the evaluation of the ethical hacking labs does not involve direct empirical data collection. Instead, the research methodology prioritizes theoretical analysis, on which the lab design is based. The comprehensive synthesis of literature from the fields of cybersecurity and ethical hacking ensures the labs' alignment with current best practices in the field.

The reliance on theoretical analysis is a direct response to the constraints of this project's scope and timeline. While the absence of empirical data collection limits direct validation of the labs' effectiveness, this approach provides a strong theoretical foundation and a structured design process. To further enhance the labs' design, future researchers could conduct empirical studies with students to assess their impact on learning outcomes, skill acquisition, and cybersecurity awareness.

3 Theoretical Framework

This chapter aims to establish the theoretical foundation for the ethical hacking labs being proposed in this thesis. It will firstly address the field of ethical hacking itself, before discussing the technical capabilities of the two devices selected for the labs, the WiFi Pineapple and the Rubber Ducky, and illustrating their role as essential tools for penetration testing within the broader spectrum of ethical hacking practices.

3.1 Ethical Hacking

Ethical hacking, often termed as penetration testing or white-hat hacking, is a critical component in the cybersecurity domain. It involves the deliberate probing

of computer systems, networks, and applications with the intent of discovering and fixing security vulnerabilities. Unlike malicious hackers, who exploit weaknesses for personal gain, ethical hackers use the same skills to improve security by identifying and addressing the flaws before they can be exploited. At its core, ethical hacking is about adopting the mindset and techniques of potential attackers to enhance security measures. [1]

Ethical hackers are authorized to breach the systems they test, but they do so under strict guidelines designed to ensure the integrity and confidentiality of the information and systems they access. This practice is governed by a code of ethics and, often, legal agreements between the hacker and the organization. [2]

3.1.1 Scope and Objectives of Ethical Hacking

Ethical hacking covers a diverse range of digital systems, from networks and websites to the physical devices that connect them, such as routers, switches, and IoT devices. This field focuses on hands-on methods to proactively enhance an organization's security posture by:

1. Identifying vulnerabilities: Ethical hackers act as simulated adversaries, searching for weaknesses that could be exploited by malicious actors. These weaknesses might be entry points for unauthorized access, gaps in security protocols, or misconfigurations in system settings. [3]
2. Security posture assessment: By attempting to breach the system with authorized permission, ethical hackers evaluate the effectiveness of existing security measures. This assessment reveals areas where defences might be inadequate and helps prioritize security improvements. [4]
3. Risk analysis: Once vulnerabilities are identified, ethical hackers analyze the potential impact of a successful attack. This risk analysis considers factors like the sensitivity of the data at stake and the potential for disruption to operations. [5]

4. Recommendations and mitigation strategies: The final step involves providing actionable recommendations on how to address the discovered weaknesses. Ethical hackers might suggest specific security patches, configuration changes, or additional security controls to mitigate the identified risks. [6]

By fulfilling these objectives, ethical hacking plays a crucial role in proactively identifying and addressing security vulnerabilities before they can be exploited by malicious actors. Furthermore, ethical hacking can be used to assess an organization's adherence to industry-specific security standards or regulations (e.g., HIPAA, PCI DSS). [3] Additionally, simulated attacks serve as powerful training tools, demonstrating the real-world impact of vulnerabilities to the staff and/or organization.

3.1.2 Ethical Considerations

Given the potential for misuse, ethical hacking is tightly bound by legal and ethical frameworks. Ethical hackers must always have explicit permission from the system owners before conducting any tests. They must also respect privacy, ensuring that any sensitive data encountered during their investigations is handled responsibly and securely. Examples of relevant laws include the Computer Fraud and Abuse Act in the U.S.A., or the GDPR (General Data Protection Regulation) in the EU. [3]

Understanding the laws surrounding technology is undoubtedly of the utmost importance; however, the law is not the end-all be-all to controlling technology. Where computers, networks, and digital information is concerned, the law is often lagging behind, and is quite slow to evolve. This is due to all the various factors and processes involved in lawmaking; in order for the law to adequately control technological security, all the factors (e.g., judges, lawyers, and policing organizations) need to be suitably informed on the matter, while also trying to create control systems that are compatible with other established parts of the law. This slow pace can hinder timely responses to new attack methods. Because the

law only provides a minimal baseline, ethical frameworks guide decision-making in ambiguous situations or in situations where technology has outgrown legal precedent. As such, it is important for ethical hackers to indeed be *ethical,* in addition to fulfilling the basic requirements of the law, thereby helping to raise the standard for the entire cybersecurity industry. [7]

3.1.3 The Ethical Hacker's Toolkit

Ethical hackers employ a wide range of tools to simulate various attack scenarios. These include, but are not limited to, network scanners, vulnerability scanners, password cracking tools, custom scripts, exploitation frameworks. [3]

Key tools utilized by ethical hackers include network scanners, notably Nmap [8] for network exploration and security auditing, and Wireshark [9] for packet analysis. Password cracking tools like Hashcat [10] are used to recover passwords by deciphering encrypted data. Frameworks such as Metasploit [11] are essential for crafting and executing exploits, providing a robust platform for security testing and many more.

The use of external devices such as WiFi Pineapple and USB Rubber Ducky, as mentioned in the introduction, illustrates the practical, hands-on approach to learning about and defending against real-world hacking techniques.

The following subheadings will delve further into the devices with which this thesis is concerned, i.e., the WiFi Pineapple and the USB Rubber Ducky. The information laid out in the respective chapters will cover the functionalities and technical capabilities of the devices; the actual labs developed using them will be discussed further into the thesis.

3.2 WiFi Pineapple: A Tool for Wireless Penetration Testing and Cybersecurity Education

The WiFi Pineapple, developed by Hak5, is a specialized device for network security testing and a popular tool in the cybersecurity world. Its power makes it

ideal for ethical hackers, security researchers, and those studying cybersecurity. [12] . The figure below depicts the two models of WiFi Pineapple that were used in the lab development.

Figure 1 WiFi Pineapple devices, standard (left) and enterprise (right) [13]

Figure 1 shows the difference between the two models of WiFi Pineapple, where the Enterprise one is larger and has more antennas. Primarily, the devices allow users to conduct advanced wireless attacks and simulations. Their capabilities will be discussed in detail in the chapters below.

3.2.1 Technical Capabilities

The WiFi Pineapple, offered in both Standard and Enterprise editions, is engineered to address a broad spectrum of wireless penetration testing needs. The Standard edition, known as the Mark VII, supports 2.4GHz wireless frequencies and can be extended to 5GHz with an additional module. This edition comes with three high-gain antennas which enhance its range and signal strength, making it suitable for basic network reconnaissance and targeted attacks. [13]

The Enterprise edition is designed for power users, cybersecurity agencies, penetration testing organizations, and enterprise applications, providing significant advancements over the Mark VII. It offers improved handling of multiple DHCP clients, capable of supporting up to 100 clients compared to the 5-10 supported by the Mark VII, and it supports 2.4GHz and 5GHz. This edition also delivers better performance in high traffic environments, capable of managing thousands of stations and base stations versus the hundreds that the

Mark VII can handle. Furthermore, it features increased backhaul throughput with Gigabit Ethernet, compared to the USB 2.0 Ethernet or 802.11n WiFi of the Mark VII, enhancing its capacity for large-scale operations.

Both editions are equipped with a network-focused System on a Chip (SoC) that manages packet analysis, attack generation, and network monitoring efficiently. They feature a web-based interface that simplifies the setup and execution of penetration tests, making these tools accessible to users with varying levels of expertise in network security. [14]

The cost-effectiveness of the Standard edition makes it accessible to individuals and educational institutions with limited budgets. In contrast, the Enterprise edition, while higher in price, is justified by its advanced capabilities and suitability for professional and large-scale cybersecurity operations.

3.2.2 Attack Vector Analysis

The WiFi pineapple excels in the following areas of network security assessment:

- Man-in-the-Middle (MitM) Attacks: It can intercept traffic between devices, allowing the analysis and potential manipulation of data. [15]
- Network Monitoring and Analysis: It gives detailed insights into network traffic, which can help identify potential threats and unauthorized access. [13]
- Rogue Access Point Creation: It allows for the easy creation of fake WiFi networks which can be used to study how attackers deceive victims. [13]

By facilitating these attack vectors, the WiFi Pineapple becomes a valuable tool in a defensive security context. It allows for the identification of vulnerabilities and potential attack paths, prompting the implementation of stronger network configurations, security protocols, and proactive threat detection strategies.

3.2.3 Defence and Mitigation Strategies

The WiFi Pineapple's wide range of attack capabilities highlights the importance of establishing strong defences and mitigation strategies to safeguard network security.

One crucial aspect is exercising caution when selecting wireless networks, particularly in public settings. Attackers frequently establish malicious networks with names deceptively similar to legitimate ones, with the intent of luring unsuspecting individuals. The presence of duplicate network names should raise suspicion, and connection should be avoided until proper verification can be conducted. [16]

Furthermore, the prioritization of HTTPS encryption plays a significant role in safeguarding data transmitted over the internet. Devices like the WiFi Pineapple can be utilized to downgrade connections from secure HTTPS to vulnerable HTTP, enabling attackers to intercept sensitive information. Vigilance in verifying the use of HTTPS protocols and the presence of valid security lock icons in the address bar is essential for ensuring connection integrity. [16]

The importance of consistently applying software and firmware updates cannot be overstated, as these patches address known vulnerabilities that devices like the WiFi Pineapple could exploit.

Device configuration adjustments can significantly enhance security. Features like enabling automatic connection to previously used WiFi networks should be disabled. Ideally, devices would be configured to forget network settings upon disconnection and avoid automatically joining open networks. This helps prevent inadvertent connections to potentially malicious networks. [16]

The use of Virtual Private Networks (VPNs) is strongly recommended, particularly when utilizing public WiFi. VPNs provide encryption of internet traffic, protecting data from interception, while also masking IP addresses. This creates an

additional layer of defence against attackers and the techniques enabled by devices like the WiFi Pineapple. Selecting a reputable VPN provider is essential for optimal security. [17]

The implementation of these strategies can greatly reduce the effectiveness of WiFi Pineapple attacks, contributing to a safer and more secure online experience for all users.

3.2.4 Educational Tool

Its application in cybersecurity education cannot be overlooked, as it provides students with invaluable hands-on experience in an educational setting. In utilizing it, students will better grasp wireless network vulnerabilities, as they can see firsthand how they can be exploited. This allows them to learn attacker techniques and develop defences against them. Furthermore, students are able to experiment and interact with the device, all within a safe and controlled lab environment.

As powerful as the WiFi Pineapple is, it is vital to adhere to these ethical principles when using it:

- Legal Compliance: Operating only within the law, obtaining necessary permissions where needed, and never causing harm to others.
- Controlled Environments: Limiting its use to authorized test environments to avoid impacting others.
- Educational Focus: Using the device to learn defensive techniques, not to compromise security or privacy.

The WiFi Pineapple is a valuable cybersecurity tool for understanding, testing, and improving wireless network security. In educational settings, it gives future professionals the skills they need to combat network threats effectively. Through responsible use, the WiFi Pineapple helps create a safer digital world, with knowledgeable ethical hackers.

The labs that were developed utilizing the WiFi Pineapple will be explained in detail in the chapter dedicated to *Designing Ethical Hacking Labs*.

3.3 USB Rubber Ducky: A HID Keystroke Injection Tool for Ethical Hacking

The USB Rubber Ducky, developed by Hak5, is a deceptively powerful cybersecurity tool. Disguised as a simple USB drive, it can simulate keystrokes and automate tasks with incredible efficiency. This makes it invaluable for ethical hackers, security researchers, and cybersecurity students. [18]. The figure below depicts a USB Rubber Ducky, illustrating its minimalistic and deceptive appearance.

Figure 2 USB Rubber Ducky

By enabling users to deploy customizable scripts with the simplicity of a few keystrokes, the USB Rubber Ducky not only enhances the arsenal of cybersecurity professionals but also serves as an excellent educational platform

to demonstrate the vulnerabilities and defences within modern computing environments.

3.3.1 Technical Capabilities

The USB Rubber Ducky may look like a typical USB drive, but it hides potent capabilities designed for the realm of ethical hacking. At its core, it uses a microcontroller to disguise itself as an HID (a Human Interface Device). This lets the Rubber Ducky slip past typical security measures that would otherwise halt unknown storage devices. By pretending to be a keyboard, it can directly run pre-coded scripts (binary files) on the target computer the moment it is plugged in, all without raising the usual red flags associated with external drives. [18]

Binary files, stored on the Rubber Ducky's microSD card, constitute the core functionality of this device. These files, generated on Payload Studio using the DuckyScript language, dictate the commands it will send, letting it automate everything from basic tasks to complex actions that carefully imitate a real user.

USB Rubber Ducky's small, ordinary appearance is a weapon in social engineering attacks. By looking harmless, it can fool people into unknowingly connecting it to their computers. This makes it a powerful tool for simulations, helping teach users and professionals how to identify and defend against these sneaky threats.

3.3.2 Attack Vector Analysis

The USB Rubber Ducky excels in the following areas of security assessment:

- Payload Delivery: It can execute pre-programmed scripts, containing potentially malicious commands, with lightning speed upon insertion into a target system. [18]
- Social Engineering: It leverages human trust in familiar devices to bypass security measures.

- Privilege Escalation: It can potentially exploit vulnerabilities to gain higher-level access within a compromised system.
- Automated Penetration Testing: It streamlines repetitive tasks and vulnerability testing. [19]

By simulating these attack vectors, the USB Rubber Ducky exposes critical weaknesses not only in technical perimeter controls but also in user awareness and the security protocols surrounding physical device handling. It emphasizes the importance of robust endpoint security measures, data encryption, and ongoing social engineering awareness training for all users within an organization.

3.3.3 Social Engineering Risks and Mitigations

Rubber Ducky's potential for misuse and malicious activity highlights how vulnerable people are to social engineering attacks and underscores the need for user education and robust security policies.
Mitigations techniques include:

- Whitelisting Known Devices: In some environments, particularly sensitive ones, it is feasible to whitelist only known and approved devices. This can be managed through software solutions that monitor and control which devices can act as input devices on a system.
- Implement third-party software designed to control device access. This helps prevent unauthorized devices from connecting to systems, adding an extra layer of security. [20]
- Raising user awareness by putting emphasis on the dangers of unknown USB devices, training of staff to verify the authenticity/safety of the device before inserting it into a computer and being aware of the possibility of physical intrusion into a secure environment.
- Policy enforcement, one of the main lines of defence for organizational structures such as companies or institutions, prohibiting the use of

unapproved USB devices, and regular audits/spot checks to help ensure compliance.

These strategies collectively form a robust defence against the threats posed by devices like the USB Rubber Ducky, aiming to reduce the risk of social engineering attacks and enhance overall security awareness within organizations. [21]

3.3.4 Educational Tool

Within cybersecurity education, the Rubber Ducky offers significant value. Students gain hands-on experience with payload creation and social engineering tactics. By directly observing the impact of keystroke injection, they develop a deeper understanding of how these attacks function. Critically, they learn how to defend against such threats. All of this occurs within a safe and controlled lab environment, fostering ethical and responsible use. [22]

Ethical guidelines are integral to training with the USB Rubber Ducky. Students learn to operate the device legally and with explicit permission, focusing on enhancing security rather than exploiting vulnerabilities. Emphasizing responsible use, the labs constantly remind students that their skills should never be used to cause harm or violate privacy. This approach not only builds their technical acumen but also instils strong ethical principles, preparing them to become cybersecurity professionals who adhere to high standards.

The USB Rubber Ducky also serves as a potent demonstration of the risks associated with physical access and seemingly harmless devices. As a tool in cybersecurity training, it empowers students to understand and mitigate these threats, contributing to a more secure digital environment through ethical and responsible use.

4 Designing Ethical Hacking Labs

In today's cybersecurity education, practical labs are essential. This chapter focuses on creating these labs with three different tools: the WiFi Pineapple, USB Rubber Ducky and an external WiFi card that supports monitor mode. These devices are chosen for their ability to effectively mimic real-world cyber threats, offering students a hands-on approach to learning. This chapter will discuss why these tools are ideal for teaching complex security concepts and the practical steps involved in setting up the labs.

The design of these labs reflects a carefully structured approach to enhancing the educational experience in cybersecurity. By incorporating the WiFi Pineapple, USB Rubber Ducky, and an external WiFi card, the labs have been crafted to provide a real-world testing environment. This chapter outlines the journey from the initial goals of demonstrating live cyber threats to the final setup of the labs. It discusses the selection of specific devices, the integration of these into the curriculum, and the emphasis on ethical practices. The overarching aim is to ensure that students not only gain practical skills but also develop an ethical mindset crucial for their future roles in cybersecurity.

4.1 WiFi Lab Development

To equip students with a comprehensive understanding of wireless network vulnerabilities, several labs have been developed using the WiFi Pineapple and an external WiFi adapter. Each lab is carefully chosen for its practical relevance and educational value, offering insights into significant cybersecurity challenges.

The first lab involves the use of an external WiFi adapter set to monitor mode, focusing on WiFi password cracking. This exercise introduces students to wireless security assessments by demonstrating how encrypted passwords can be cracked using tools like Aircrack-ng. It underscores the necessity of robust security practices in safeguarding wireless networks.

In another lab, students engage with the WiFi Pineapple to create an Evil Portal. This setup simulates a common attack where malicious access points mimic legitimate ones to deceive users and harvest their data. The lab aims to teach the critical importance of verifying network authenticity and the risks associated with connecting to unknown WiFi networks.

Lastly, the DNS Spoofing lab allows students to manipulate DNS requests using the WiFi Pineapple, redirecting users to fraudulent websites. This lab helps students understand the mechanics behind DNS attacks and the potential for such exploits to compromise network security. It also discusses strategies to mitigate these risks, emphasizing proactive defence measures.

In conclusion, these labs start with the basics and gradually get more advanced. Students begin with simple password cracking before tackling complex challenges like setting up an Evil Portal and manipulating DNS records. This structured approach builds skills, confidence, and provides a comprehensive network security education.

4.1.1 Wi-Fi Cracking

A strong WiFi password serves as the first line of defence for a wireless network. In an era where personal information and online activities are constantly under threat, setting a robust password is paramount. Unfortunately, many individuals opt for simple, easily guessed passwords such as birthdates or common words. This practice leaves their WiFi networks highly vulnerable to cracking attempts. Using readily available hacking tools and custom wordlists, malicious actors can often compromise weak passwords with alarming ease, potentially leading to data breaches, unauthorized network access, and further security risks.

This chapter investigates WiFi security vulnerabilities by using hands-on ethical hacking techniques. Specifically, the focus is on demonstrating WiFi password cracking using an external WiFi card in monitor mode and the Aircrack-ng suite.

The purpose of this exploration is to emphasise the importance of understanding and mitigating common wireless network weaknesses.

This experiment relies on a few essential tools and files. Firstly, an external WiFi card that supports monitor mode is required. During the development of this lab the CtrlFox Atheros AR9271 was used (see Figure 3). [23]

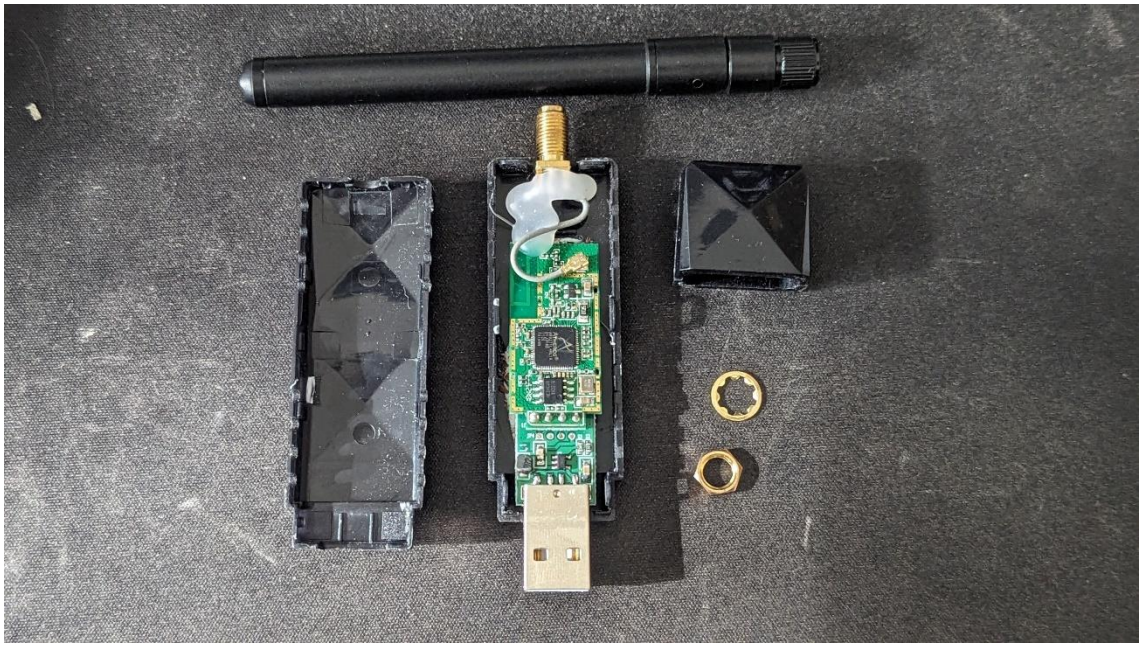

Figure 3 External Wi-Fi adapter that supports monitor mode.

The monitor mode allows the card to passively capture and analyze all wireless traffic within range, even if the traffic is not intended for the user's device. Secondly, the Aircrack-ng suite, a collection of tools specifically designed for wireless network security assessments, was used. Lastly, to conduct the password cracking demonstration, a custom wordlist was created. This wordlist contains dates from January 1, 1930, to December 31, 2024, in both DDMMYYYY and MMDDYYYY formats. A Python script (date_wordlist_generator.ipynb) was used to automatically generate this wordlist.

The WiFi cracking process adhered to a series of structured steps. First, to minimize potential interference from other processes, those that could impact the experiment were terminated using the "airmon-ng check kill" command. Next, the "iwconfig" command was used to identify the name of the wireless interface.

Monitor mode, crucial for capturing all network traffic, was then activated on this interface. Using the "airodump-ng" command, the surrounding WiFi networks were scanned. A target network was selected based on its signal strength, encryption type, BSSID, and channel information. To obtain an authentication handshake, "airodump-ng" was utilized once more, specifying the target network's BSSID, channel, and a designated file to store the captured data. This captured handshake is the key to cracking the network's password.

Optionally, to expedite the handshake capture process, a connected client could be forcibly disconnected from the target network. This was achieved by issuing a deauthentication attack using the "aireplay-ng" command.

Finally, Aircrack-ng attempted to crack the network password using the captured handshake and the provided custom wordlist. A successful crack would reveal the network's password. After the experiment, network services were restarted, and the WiFi card was returned to its normal operational mode. The screenshot below shows the successful cracking of the Wi-Fi password.

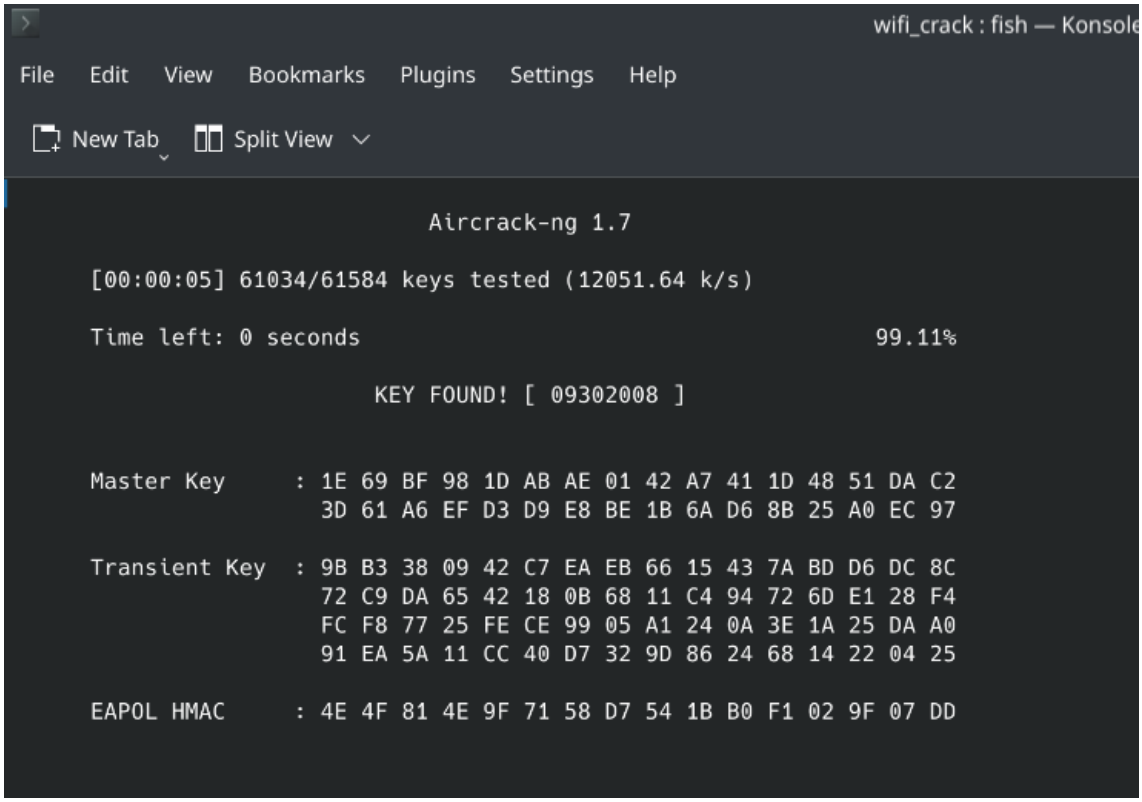

Figure 4 Successful cracking of the Wi-Fi password using Aircrack-ng 1.7

Given the context of the experiment, where students are informed that the target password is a birthdate, a specialized wordlist was essential for maximizing the chances of a successful password crack. A Python script was used to generate a comprehensive list containing all potential birthdates from January 1st, 1920, to December 31st, 2024. This date range was selected to realistically reflect the age distribution of individuals likely to be setting up WiFi networks. Both DDMMYYYY and MMDDYYYY formats were included to account for variations in how people might represent their birthdates. To further streamline the wordlist, a secondary script was employed to eliminate duplicate entries. For example, a date like "05101985" (representing October 5th, 1985) could be written in both DDMMYYYY and MMDDYYYY formats (representing May 10th, 1985), leading to redundancy. Removing these duplicates increases the efficiency of the password cracking process. The wordlist with duplicates contains 76,704 entries, while the cleaned wordlist contains 61,584 entries. While the difference in password cracking time between the two wordlists might seem small—just one second—it is important to remember that every second counts in large-scale operations. The cleaned wordlist takes 5 seconds (as seen in Figure 4), and the duplicate-filled one takes 6 seconds. Emphasizing best practices by reducing redundant processes is crucial for efficiency and effectiveness, and because of this, there is a bonus point for the lab if the student generates the wordlist that does not contain duplicate entries themselves.

Finally, the students must write a report (of up to 400 words) explaining their approach and providing proof of completion in the form of screenshots. If the student aimed for the bonus point, they would have to also include the code they used to generate the wordlists and explain how they ensured there is no duplicate entries.

4.1.2 Evil Portal Attack

An "evil portal" is a common wireless attack technique that intercepts and manipulates web traffic. To illustrate this concept, a project was undertaken using

a Pineapple Enterprise device. The demonstration required a Pineapple Enterprise and a router with internet access.

The first step involved preparing the Pineapple device. The WiFi Pineapple was connected to a router via WAN cable to provide internet access, and the initial setup of the device (name, password, network settings) was completed. Next, an open access point (AP) titled "FreeWifi2.4gz" was created to attract unsuspecting users. Then, the Evil Portal module was installed on the Pineapple device.

To capture user data, a deceptive login page was designed. For compatibility with the Evil Portal module, a PHP-based login page (evil_portal_metropolia) was used.

The goal of the lab was to mimic the oma.metropolia.fi login page as an evil portal which would fetch the information the user inputs, and then redirect them to the internet, making the login page seem legitimate. The idea is to mimic the login page as closely as possible including the fonts, colours, opacity, and to make this compatible for both desktop and mobile. Figure 5 shows the desktop version of the cloned site, whereas Figure 6 depicts the mobile version.

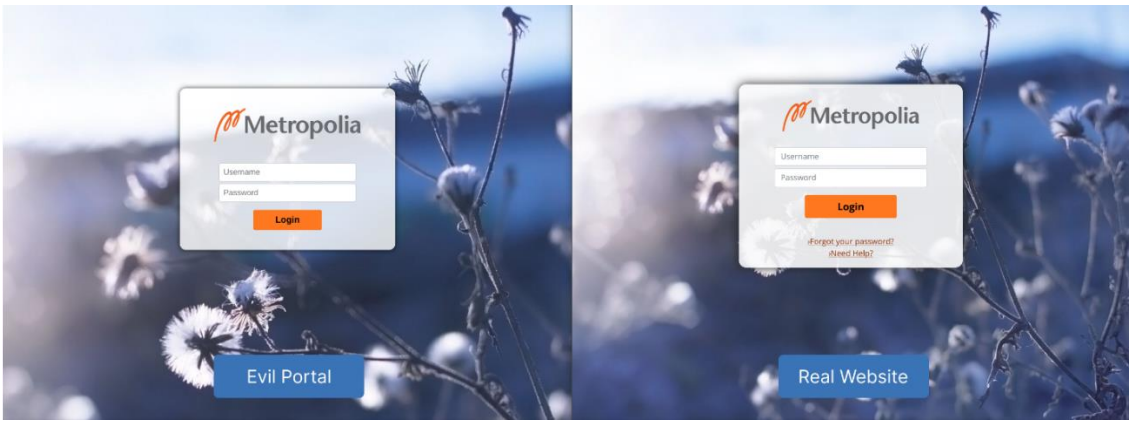

Figure 5 Desktop version of the cloned site

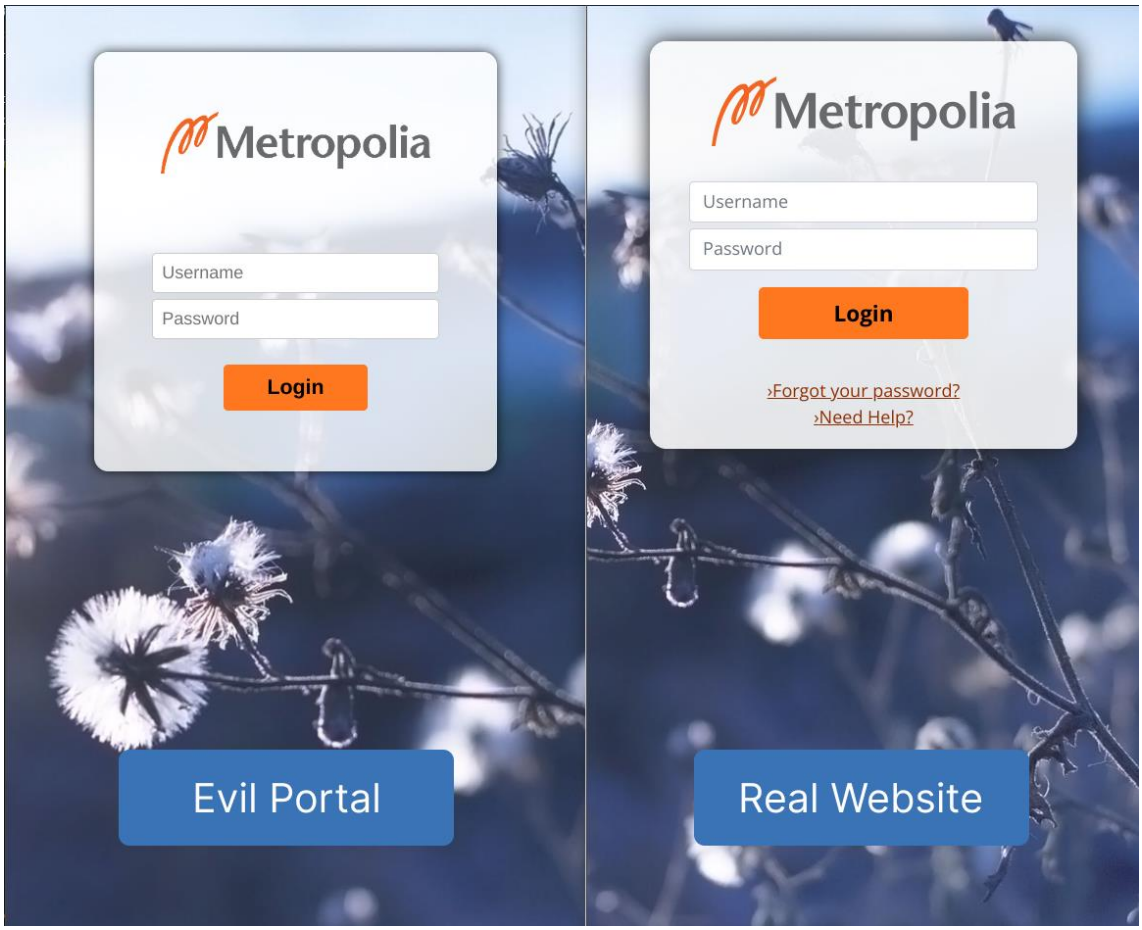

Figure 6 Mobile version of the cloned site

As the figures above show, the mimicry is close but not perfect. The main difference is the intentional removal of the 'Forgot your password?' and 'Need Help?' links to avoid raising suspicion for users who might click them. Other subtle differences include variations in the opacity of certain elements and a slight misalignment of the background image. The goal is not to create a perfect clone of the page, but rather a convincing enough imitation.

For the deliverables, the student has to detail the process of configuring the Open AP and installing the evil portal, explain the steps taken to customize and mimic the target website and finally show screenshots of both the desktop and the mobile version as well as a live demo to the professor.

4.1.3 DNS Spoofing Attack

DNS spoofing is a technique that manipulates web traffic by redirecting users to fraudulent websites. To gain a deeper understanding of this threat, a DNS spoofing attack was configured using a WiFi Pineapple. Firstly, the necessary "DNS Spoof" module was installed on the Pineapple Enterprise. Within this module's settings, a target website for imitation (e.g., www.example.com) and the IP address of a controlled, fake website were specified. [24]

Here is a breakdown of how a DNS spoofing attack works: when a user on the compromised network attempts to visit a legitimate website, the WiFi Pineapple intercepts the DNS request that is responsible for resolving domain names into IP addresses. Instead of providing the legitimate IP address, the Pineapple sends a malicious one – the IP address that was previously configured. This causes the user's browser to unknowingly connect to the attacker-controlled server.

DNS spoofing poses significant dangers. Fake websites can be designed to closely mimic legitimate ones, tricking users into divulging sensitive information (phishing). An attacker can use redirection to distribute malware to the victim's device. Additionally, this technique enables an attacker to monitor the victim's internet activity.

Similar to the previous lab, the students have to create a website that closely mimics oma.metropolia.fi and host it locally. This website should then be configured to the spoofed domain so when the users connect to oma.metropolia.fi it will redirect them to the local spoofed version.

As for the deliverables, the student has to write a report (of up to 600 words) to outline the steps taken to set up the DNS spoofing attack, describe their efforts on mimicking the oma.metropolia.fi page, and provide screenshots of the cloned site running on the local server.

4.1.4 Review of WiFi Pineapple Lab Design

WiFi Pineapple labs are a key part of practical cybersecurity training, exposing students to advanced network hacking techniques. With structured labs on WiFi Cracking, Evil Portal, and DNS Spoofing, students learn the tools and methods real-world attackers use. This direct, hands-on approach is vital for deeply understanding network weaknesses and how they can be exploited.

Each lab has clear goals to test students' ability to apply their knowledge. One of the exercises offers a bonus point for those students who go beyond the basics, encouraging deeper learning and research. This is meant to not only improve the students’ skills, but also motivate them to dive deeper into the subject.

The labs clearly explain the ethical responsibilities that come with penetration testing. Students learn that ethical behaviour is important in cybersecurity and that their skills should be used to strengthen security, not cause harm.

The WiFi Pineapple itself is a powerful tool that goes beyond penetration testing. It can be used for advanced scanning, deauthentication attacks, and creating honeypots – systems designed to lure and study attackers. The labs highlight this versatility, preparing students to use a wide range of cybersecurity tools in the future.

By completing the WiFi Pineapple labs, students gain more than just practical skills for handling real-world cybersecurity threats. They build a strong ethical base and understand why security must be proactive. These labs ensure that as students progress, they are ready to tackle complex security problems and help make the digital world safer.

4.2 USB Rubber Ducky Lab Development

To introduce students to the concept of USB Rubber Ducky device, a series of exercises and labs were developed. These labs were designed with real-world

relevance in mind, teaching students the basics of these devices, their potential for malicious use, and methods for preventing such attacks. Given the emphasis on real-world applicability, the choice to implement Windows 11 on the target device was influenced by Windows' market leadership in desktop operating systems, with a 79.16% share in Finland, as can be seen in the graph below

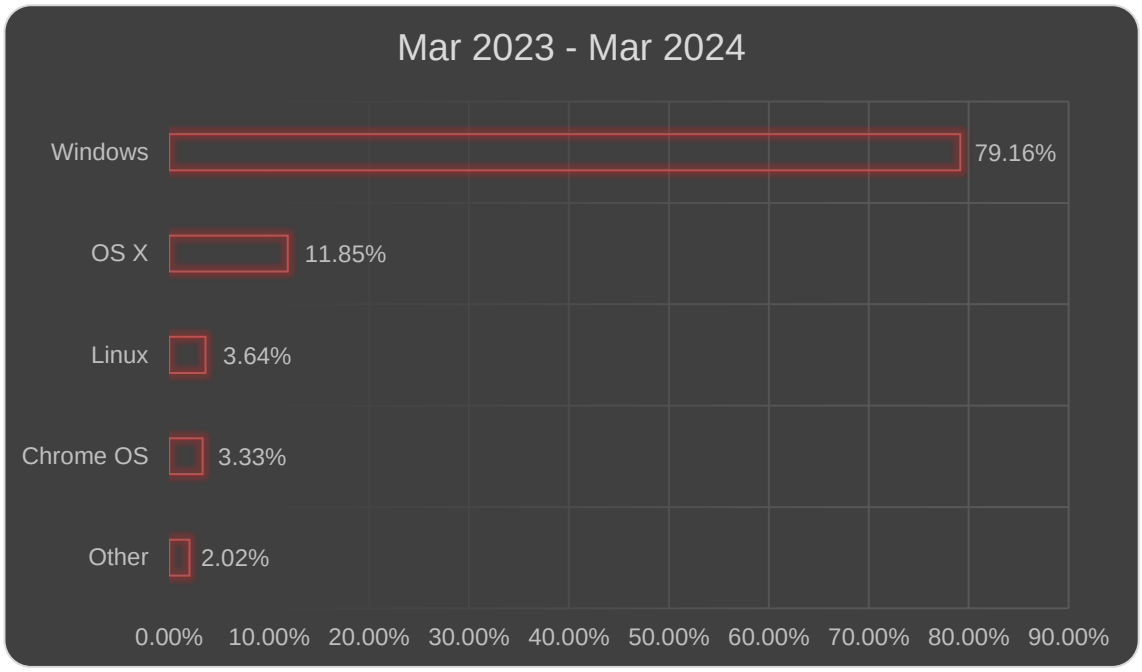

Figure 7 Desktop OS market share in Finland. [25]

While most of the labs do not require a specific operating system for the host machine, one of them does require a Linux distribution of the students' choice. The chapters below will cover the development and content of these labs, all of which are based on the use of USB Rubber Ducky. In addition, there will be a discussion on the attempted but ultimately scrapped fourth lab titled 'Browser Saved Password Fetcher,' alongside the successfully developed labs:

1. Last Picture Taken (Android)
2. Reverse shell attack
3. Documents extraction

The labs were made to increase in difficulty, the first one being a simple code to fetch the last picture taken from an Android phone, and the last one being more complex, as it involves generating a .bat script that the students have to run to fetch the specified filetypes from the target machine.

4.2.1 Android Document Fetcher (Android)

In order to give the students an opportunity to get acquainted with the device, this lab was relatively straightforward. The objective is to copy the documents from the target device to the USB Rubber Ducky. The Android phone required for this exercise will be provided by the professor and importantly unlocked. The idea is for the student to learn the basics of DuckyScript. There are many ways this exercise can be done, so it is up to the students to find the optimal way. This exercise was allocated two points: one point for successful completion of the task and a bonus point if the student manages to fetch the picture within 15 seconds. As for the deliverables, the student is to document the process either by screen recording, or if they are aiming for the bonus point, by using an external camera to record the target phone alongside a chronometer (physical, app-based or on a website) for accurate timekeeping like in the figure below.

Students are expected to write a report (up to 400 words) detailing their

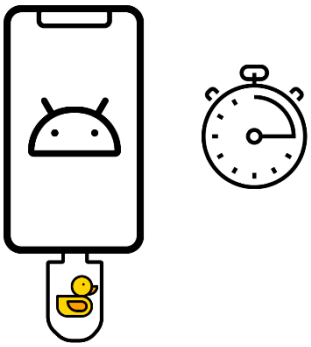

Figure 8 Recording instructions.

optimization approach, the code used, and the results achieved. This report should be submitted along with the recording.

4.2.2 Reverse Shell Attack

Following the first lab, the students should have a basic understanding of DuckyScript and its utilities. The second lab expands on the first by asking

students to create a reverse shell from the target machine to the host machine. This lab involves two machines (the target machine running Windows 11, and the host machine running a Linux distribution of the students' choosing) and the USB Rubber Ducky. While there are various scripts online for performing this attack, the lab is a good exercise for customizing and adapting these scripts, so they can better understand the underlying principles of such an attack. Again, the Windows 11 operating system was used in this case due to its large market share, both worldwide and in Finland.

While there are many ways to perform this attack, during the development of the lab, the Metasploit framework was utilized. The latter is a powerful tool designed for penetration testing and vulnerability research. [11]

Following this setup, the students are expected to create a payload on the host machine, which will be fetched and run by the target machine later on with the help of USB Rubber Ducky.

At last, the student has to write a short report (of up to 500 words) demonstrating the access gained with a screenshot, describing the tools and the commands/code they wrote and if they want to go for a bonus point then they have to establish a persistent reverse shell connection and explain how they did this.

4.2.3 Document Extraction

The last lab is creating a script for USB Rubber Ducky that will extract all .docx, .txt and .xlsx documents from the target machine's Downloads, Documents and Desktop folders (as well as all of their subfolders) to the USB Rubber Ducky. This lab shows a great way to introduce students to writing Windows batch files (.bat).

Any kind of documents can be extracted with the script, but for simplicity's sake it was done for .docx, .txt and .xlsx. formats only. First, a simple script was developed using DuckyScript which was able to copy the files mentioned above

to the USB Rubber Ducky without utilizing a .bat file. The code snippet below is an example of a script that fetches the documents from the target's desktop.

```
ATTACKMODE HID STORAGE
DELAY 3000
GUI r
DELAY 1000
STRING cmd
DELAY 400
ENTER
DELAY 1000
STRING for /d %d in (C:\Users\PC\Desktop\*) do for %x in (%d\*.txt, %d\*.docx,
%d\*.xlsx) do copy "%x" D:\files
DELAY 500
ENTER
```

Listing 1. A DuckyScript code that fetches documents.

Listing 1 iterates over all directories on the user's desktop. For each directory, it looks for files with the extensions .txt, .docx, and .xlsx. Each file that matches these criteria is then copied to the *D:\files* directory, i.e., the USB Rubber Ducky.

The issue with the script above occurred when trying to copy files from the Downloads and Documents directories after copying the Desktop directory; the delay was unpredictable and would be based on the amount of documents that can be found in those directories (e.g., if the delay was set to 10 seconds, but the process took longer, it would break the script), so having a static delay would potentially break the script. The solution was a Windows batch file (.bat) which would be planted on USB Rubber Ducky's storage and the script would run that .bat file instead.

Initially a .bat file was created which copies .txt, .docx, and .xlsx files from the user's Desktop, Documents and Downloads folder, identifies each qualifying file within that folder (including subfolders), and then copies these files to a specified destination directory. The script was then further optimized.

To make the extraction more orderly, the students are advised to utilize '*xcopy*' or '*robocopy*'. Moreover, these tools are also a faster way to fetch the files.

Robocopy (Robust File Copy) is a command-line tool in Windows for file copying and directory synchronization. One of its key features is the ability to utilize multiple logical processors (/MT switch), potentially speeding up large file transfer operations as can be seen in Figure 9.

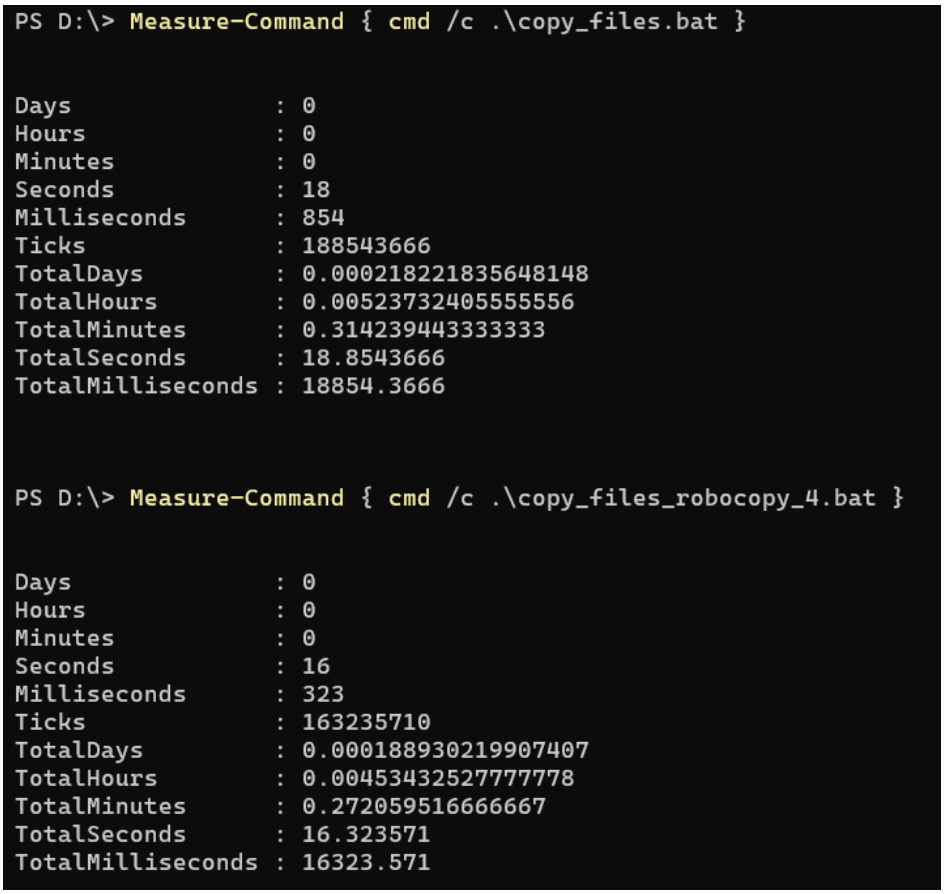

Figure 9 Time measurement of the .bat script with copy and robocopy.

The figure above shows the comparison between the runtime of the script with the ‘copy’ command and the one with the ‘robocopy’ command. When using robocopy with 4 logical processors the code executes 13.42% faster than with the normal copy command. Furthermore, when using the ‘robocopy’ command to fetch the files, it copies the full path from the target machine to the USB Rubber Ducky (e.g., if a file was located on the target machine under Desktop/test_folder/file1, robocopy will create a Desktop/test_folder/file1 inside the destination folder of the USB Rubber Ducky.)

The process carries a risk of device malfunction. To mitigate this, a system restore point named 'Lab3Setup_restore_point' has been created. If the restore fails, students should reinstall Windows 11 and then run the LabSetup.bat script. This script sets up the necessary file and folder structures, allowing instructors to easily verify if students have achieved the lab's objective.

Finally, the students are required to write a report (of up to 600 words) to describe the code and the commands they used, the outcome documented with screenshots, and if they wish to get a bonus point, they need to modify the script, so it doesn’t fetch any files larger than 2MB.

4.2.4 Browser password fetcher

Initially an attempt was made to create a lab that fetches saved browser passwords (without having to open the browser itself). However, while developing the lab, it became apparent that it would be too difficult for the students, and not as straightforward as it was initially thought.

The lab involved first identifying the location of the browser password file. Chrome was selected due to its dominant market share worldwide, but mostly in Finland with a 68.73% of desktops in Finland (Mar 2023 – Mar 2024) having it installed as the figure below depicts.

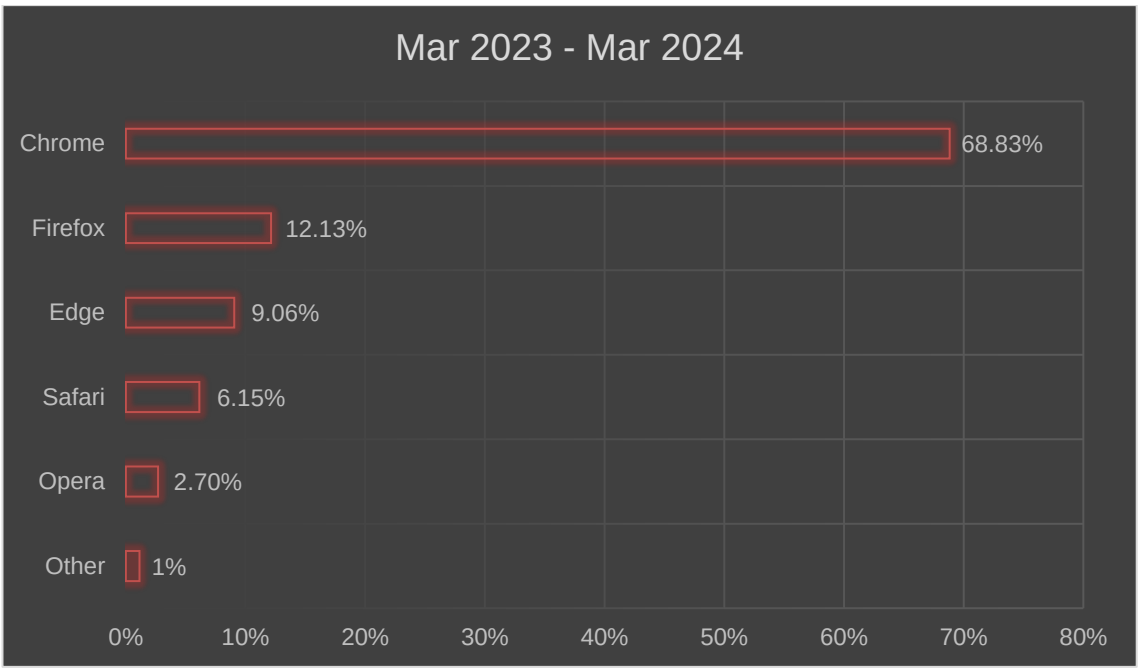

Figure 10 Desktop browser market share in Finland [26]

Firstly, the students had to locate the directory where Chrome's password file was stored. By default, that would be C:\Users*[username]**AppData**Local**Google**Chrome**User**Data**Default**Login Data.* After copying the Login Data file in a folder/USB, the next step – the challenging step – is the decryption of the passwords. The file requires Data Protection API (DPAPI) and the master keys to decrypt it in another machine. While the SID was accessible and copyable (found on '*C:**Users**[Username]**AppData**Roaming**Microsoft**Protect**[SID]*\'), the problem came from the '*SYSTEM*' and '*SAM*' files which are found on `*C:*\*Windows*\*System32*\*config*` as they cannot be copied because they are currently in use when the Windows is on. The workaround would be to live boot an OS to extract them. This was out of the scope of the lab; thus, it was dropped, and the task was not integrated into the lab.

4.2.5 Review of USB Rubber Ducky Lab Design

The USB Rubber Ducky labs are a vital part of the cybersecurity curriculum, highlighting the importance of physical penetration testing. Students learn how to use the USB Rubber Ducky to create and run scripts effectively. They don't just follow instructions but grasp the concepts behind keystroke injection attacks – a common threat in real-world security breaches. These hands-on exercises show how ordinary-looking devices can be powerful cybersecurity tools.

Each lab includes a set of deliverables that require students to demonstrate their ability to apply what they've learned. To encourage deeper exploration and mastery of the content, an additional bonus point is available for those who exceed the basic requirements of the lab. This incentive motivates students to delve further into the technical aspects and potential applications of the skills they are acquiring.

Ethics are a cornerstone of the learning process and covered clearly in the lab instructions. Students learn about legal limits and responsible behaviour in penetration testing. The emphasis is on using these skills to strengthen security, not on exploiting weaknesses.

By completing these labs, students gain a well-rounded understanding of physical penetration testing – both the technical and ethical sides. They leave knowing how tools like the USB Rubber Ducky can be used in ethical hacking to improve organizational security. This experience prepares them to apply their skills in real-world situations, protecting against and lessening cybersecurity threats.

5 Conclusion

Practical labs focusing on the WiFi Pineapple and USB Rubber Ducky devices have successfully been developed in this project, delivering a hands-on approach to cybersecurity education that emphasizes problem-solving and ethical hacking practices.

The labs were designed to cultivate critical thinking and adaptability, encouraging students to seek out additional information and tackle problems in environments where complete data may not always be readily available. The choice of this thesis topic provided a unique opportunity to explore and master new technologies, fostering a learning environment where curiosity drives exploration and understanding.

The primary goal of these labs is to familiarize students with practical tools used in cybersecurity, teaching them to apply these tools in contexts that mimic real cybersecurity threats. This approach not only enhances technical proficiency but also deepens students' understanding of the strategic aspects of potential attackers—a crucial skill for effective defence.

Despite the comprehensive design of the labs, the scope and timing of this study introduced limitations, notably the absence of empirical testing with students. This testing is essential for evaluating the labs' effectiveness and identifying areas for improvement. Future refinements, guided by feedback from users and overseen by educators, will ensure that the labs continue to meet educational needs effectively.

The hands-on experience provided by these labs is expected to significantly enrich the learning process in cybersecurity education. By engaging directly with hardware and software tools that simulate real-world security challenges, students gain a profound understanding of system vulnerabilities and defensive strategies. This proactive learning environment not only prepares students to use

these tools but also to think critically and ethically about solving security problems in their future careers.

In conclusion, this thesis establishes a dynamic foundation for learning environments where students actively engage with real-world tools and scenarios, significantly enhancing their educational journey in cybersecurity. The ongoing evolution of these labs is vital for keeping pace with the rapidly evolving cybersecurity field, ensuring that future students are well-equipped to meet the challenges of the digital age.

References

1. Occupy The Web. Linux basics for hackers: Getting started with networking, scripting, and security in Kali. 1st ed. 6 , editor. San Francisco: William Pollock; 2019.
2. OWASP Foundation. OWASP Code of Conduct. [Online].; 2020 [cited 2024 April 4]. Available from: https://owasp.org/www-policy/operational/code-of-conduct.
3. Cisco Learning Network. Cybersecurity Essentials Course [Networking Academy: Cybersecurity Essentials Course]. [cited 2024 April 3].
4. Garry Kranz LRMC. Tech Target. [Online].; 2021 [cited 2024 April 4]. Available from: https://www.techtarget.com/searchsecurity/definition/ethical-hacker#:~:text=The%20purpose%20of%20ethical%20hacking,other%20malicious%20activities%20are%20possible.
5. CompTIA. CompTIA. [Online]. [cited 2024 April 2]. Available from: https://www.comptia.org/content/articles/what-is-ethical-hacking#:~:text=Ethical%20hackers%20are%20tasked%20with,that%20protect%20organizations%20from%20attacks.
6. OWASP Foundation. OWASP Testing Guide. [Online].; 2014 [cited 2024 April 4]. Available from: https://owasp.org/www-project-web-security-testing-guide/assets/archive/OWASP_Testing_Guide_v4.pdf.
7. Charles P. Pfleeger SLPLCk. Security in Computing: Addison-Wesley Professional; 2023.
8. Nmap. Nmap Reference Guide. [Online]. [cited 2024 April 2]. Available from: https://nmap.org/book/man.html#man-description.
9. Wireshark Foundation. Wireshark. [Online]. [cited 2024 April 11]. Available from: https://www.wireshark.org/about.html.
10. Hashcat. Hashcat. [Online]. [cited 2024 April 8]. Available from: https://hashcat.net/wiki/.

11. Metasploit. Metasploit. [Online]. [cited 2024 April 1]. Available from: https://www.metasploit.com.
12. Hak5. Hak5. [Online]. [cited 2024 April 1]. Available from: https://shop.hak5.org/products/wifi-pineapple.
13. Hak5. Hak5 WiFi Pineapple Features. [Online]. [cited 2024 April 1]. Available from: https://shop.hak5.org/pages/wifi-pineapple-features.
14. Hak5. Hak5 WiFi Pineapple Enterprise. [Online].; 2023 [cited 2024 April 1]. Available from: https://docs.hak5.org/wifi-pineapple-enterprise.
15. Mulaj D. System Weakness. [Online].; 2022 [cited 2024 April 14]. Available from: https://systemweakness.com/wifi-pineapple-and-mitm-attacks-c47bd4ade470.
16. GeeksForGeeks. GeeksForGeeks. [Online].; 2022 [cited 20234 April 4]. Available from: https://www.geeksforgeeks.org/how-to-defend-against-wi-fi-pineapple/.
17. Lutkevich B. Tech Target. [Online].; 2022 [cited 2024 April 14]. Available from: https://www.techtarget.com/searchsecurity/definition/Wi-Fi-Pineapple#:~:text=Use%20a%20virtual%20private%20network,read%20the%20data%20being%20transmitted.
18. Hak5. Hak5 USB Rubber Ducky. [Online]. [cited 2024 April 1]. Available from: https://docs.hak5.org/hak5-usb-rubber-ducky.
19. Arthur Salmon WLMM. Applied Network Security. 1st ed. Birmingham: Packt Publishing; 2017.
20. al. SNe. USBlock: Blocking USB-based keypress injection attacks. Austria: SBA Research; 2018.
21. Kahmen J. Turing Point. [Online].; 2023 [cited 2024 April 2]. Available from: https://turingpoint.de/blog/awareness-mit-einem-usb-rubber-ducky/.
22. National Inventors Hall of Fame. Invent. [Online]. [cited 2024 April 4]. Available from: https://www.invent.org/blog/trends-stem/hands-on-STEM-learning.
23. AliExpress. [Online]. [cited 2024 April 1]. Available from: https://de.aliexpress.com/item/32652651808.html?spm=a2g0o.order_list.order_list_main.65.56b21802FCJyHP&gatewayAdapt=glo2deu.

24. Mike Chapple JMSDG. Certified Information Systems Secyruty Professional Official Study Guide. 9th ed. Hoboken: Sybex; 2021.
25. Stat Counter Global Stats. StatCounter. [Online].; 2024 [cited 2024 April 9]. Available from: https://gs.statcounter.com/os-market-share/desktop/finland.
26. Stat Counter Global Stats. Stat Counter. [Online].; 2024 [cited 2024 April 2]. Available from: https://gs.statcounter.com/browser-market-share/desktop/finland/#monthly-202303-202403-bar.

WiFi Pineapple Lab

This lab focuses on the WiFi Pineapple, a device that is specialized for network security testing and is a popular tool in the cybersecurity world. You will get hands-on experience using the WiFi Pineapple for task automation and explore its capabilities. Importantly, we will also cover defensive strategies to protect yourself from potential WiFi attacks. Additionally an external WiFi adapter that supports *Monitor* mode will be used to crack WiFi passwords. By the end, you will have a well-rounded understanding of these unique tools.

Setting up your WiFi Pineapple

It is important to always connect the WiFi antennas before powering on your WiFi Pineapple! The easiest way to connect your WiFi Pineapple to your device is through a USB-C cable. For macOS, additional steps are needed and can be found here.

Once connected you can access the WiFi Pineapple Stager at http://172.16.42.1:1471 from which you can find step by step instructions on how to proceed. If you need more information, check the WiFi Pineapple Mark VII documentation.

Figure 1 WiFi Pineapple Mark VII

Familiarize yourself further with online resources. You will find detailed information in their documentation, tons of helpful videos online, and you can even ask the LLM of your choosing for additional insights.

Important Disclaimer: Ethical Hacking and Responsible Practice

The techniques and tools explored in this lab are intended solely for educational purposes within a controlled environment. Using these techniques to interfere with networks you don't own, or to access systems without explicit permission, is illegal and unethical.

Cybersecurity is a complex field where actions have real-world consequences. Exercise responsible judgment, always obtain consent, and prioritize the security of others. Here are some key points:

- Obtain Permission: Never attempt to hack into a system or network without clear authorization from the owner. Doing so, even with good intentions, can have severe legal and ethical ramifications.
- Respect Boundaries: Focus your skills on your own authorized lab environments or simulated targets.
- Stay Within the Law: Be familiar with cybersecurity laws and regulations in your area. Violating them carries serious consequences.
- Protect Privacy: Handle any sensitive information you may encounter with the utmost care.

Ethical hacking is a powerful tool for improving cybersecurity. Use this knowledge to build defenses, not to cause harm!

Part I: Crack WiFi Password (1p + 1p bonus)

Difficulty: Easy

Objective: Learn to crack a WiFi password using an external WiFi adapter.

Setup:

1. External WiFi Adapter: An essential tool for this lab is an external WiFi adapter capable of monitor mode. This functionality is crucial for observing and capturing WiFi communications.
2. Choose a Linux distribution of your liking as your operating platform.
3. The Password Cracking Tool: Install the Aircrack-ng suite or any other password cracking tool on your Linux system. Aircrack-ng is comprehensive toolkit designed for assessing WiFi network security.

Your goal is to hack into a WiFi network named “HackMe.” The network's password is known to be a birthdate, ranging anywhere from 1930 to 2024. The format of the date is either DDMMYYYY or MMDDYYYY.

Bonus Point (1p): - Earn a bonus point by creating your own wordlist for the password cracking. Make sure to not have any duplicate entries from your list.

Task: Write a report (up to 400 words) that includes:

- A brief description of your approach.
- Screenshots showing the successful password crack.
- A list of commands you used.
- If you went for the bonus point, include:
    - The code you wrote to generate the wordlist.
    - An explanation of how you created the wordlist, the programming language used, and how you ensured there were no duplicates.

Part II: Evil portal (4p)

Difficulty: Medium/Hard

Objective: Set up an "Evil Portal" on the WiFi Pineapple device to simulate oma.metropolia.fi

Equipment:

- WiFi Pineapple
- Router with internet access

Steps:

- Open AP Configuration:

  Create an Open AP. This will serve as the gateway for capturing traffic, simulating a free WiFi hotspot. Ensure the AP is unencrypted to mimic real-world public WiFi vulnerabilities.

- Evil Portal Module Installation:

  Install the Evil Portal module on the Pineapple device. This allows for the creation of a fake login page aimed at capturing user credentials and other sensitive information.

- Login Page Setup

  Use the `evil_portal_simple_login` to set up a simplified login page. This version is designed to capture essential user credentials (username, password) and device information (MAC address, IP address, hostname).

  Clone the repository and follow the installation steps provided on the project's GitHub page.

- Design it to match oma.metropolia.fi login page.

  Go for accuracy in the layout, colors, fonts, opacity, and overall design to make it as convincing as possible.

  Make it compatible for both Desktop and Mobile

  Note: The background image on oma.metropolia.fi changes seasonally. Match the current background image present at the time of your project to maintain authenticity.

- Example Outcome
  - Below are examples of the final Evil Portal appearance,

    showcasing the results for both desktop and mobile versions.

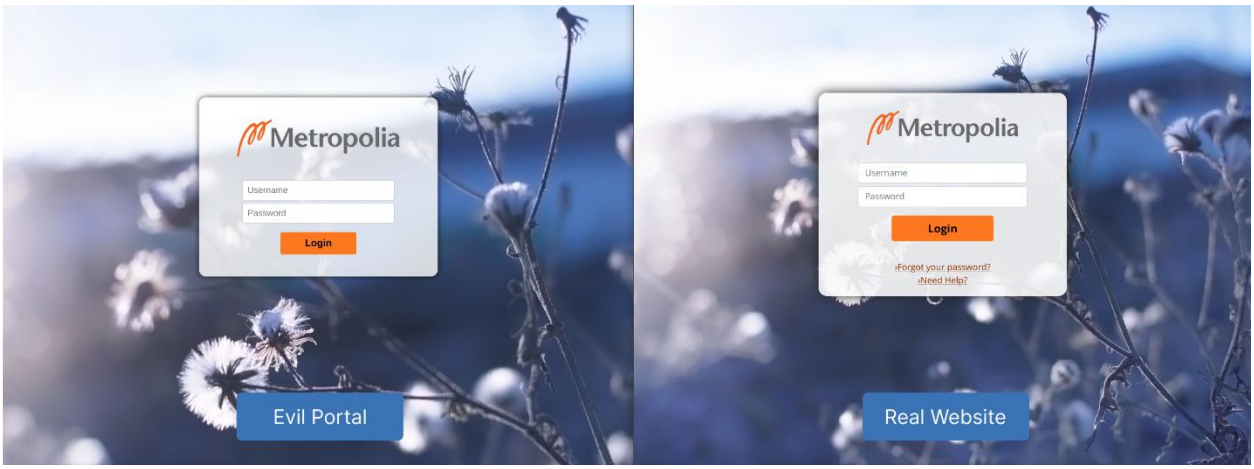

Figure 1 Desktop Version

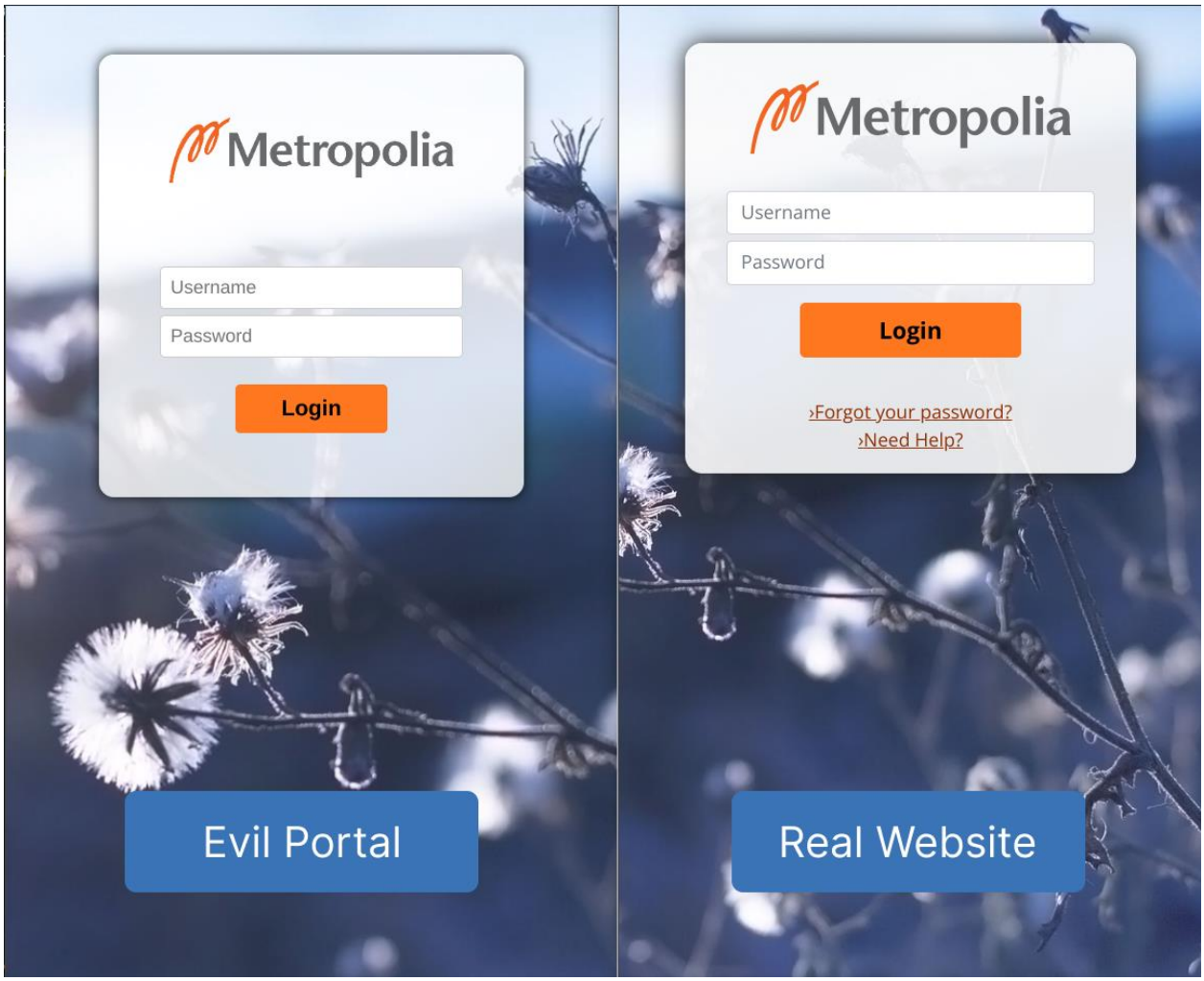

Figure 2 Mobile Version

As you can observe, while the mimicry is close, it is not perfect. Notable differences include:

- The placement of the background picture is slightly off.
- The opacity of certain elements isn’t matched 100% with the original.
- The "Forgot your password?" and "Need help?" links are intentionally removed to avoid alarming anyone who might press them etc.

Task: Write a report (up to 500 words) on:

- Detail the process of configuring the Open AP and installing the Evil Portal module.
- Describe the steps taken to set up and customize the login page, especially your efforts to mimic the oma.metropolia.fi login page design for both desktop and mobile compatibility.
- Include screenshots showing the final appearance of your Evil Portal, demonstrating both desktop and mobile versions, and show the live demo to your professor.
- Reflect on the ethical implications of deploying such technologies and how they can be mitigated in real-world scenarios.

Part III: DNS Spoofing (4p)

Difficulty: Hard

Objective: Simulate a DNS spoofing attack to redirect traffic from oma.metropolia.fi to a locally hosted clone of the site.

Equipment and Prerequisites:

- WiFi Pineapple
- A local PC to host the cloned website
- Knowledge of web development (HTML, CSS, backend technology)

Steps:

- SSH Connection to WiFi Pineapple:
  - Establish an SSH connection to your Pineapple device with the command: ` ssh root@172.16.42.1`
- Modify the Hosts File on Pineapple:

    - Access and edit the `/etc/hosts` file to map the oma.metropolia.fi domain to the IP address of your local server hosting the cloned website.
    - After editing `/etc/hosts`, apply the changes by restarting the DNS service on the device with the command: `killall dnsmasq && /etc/init.d/dnsmasq start`
- Create a Local Clone of oma.metropolia.fi:
    - Develop a local webpage that closely mimics the oma.metropolia.fi login page. Use web development skills to replicate the site's HTML, CSS, and any required backend functionality. If you completed the previous lab, you might reuse and run the CSS and HTML code from that lab locally (see figure 2 and 3 as an example and read the paragraph after the figures).
    - Ensure your cloned site is ready to capture or log the input data.
- Connect the Local Server to the Spoofed Domain:
    - Configure your local web server to host the cloned page. Make sure the server is running and accessible through the IP address you've mapped in the Pineapple's `/etc/hosts` file.
    - When users connected to the Pineapple's network attempt to visit oma.metropolia.fi, they should now be redirected to your local spoofed version of the site.

Task: Write a report (up to 600 words) to describe the process:

- Outline the steps taken to set up the DNS spoofing attack and clone the website.
- Discuss any technical challenges encountered and how they were addressed.
- Reflect on the effectiveness of the spoofed page in mimicking the original and the ethical implications of such attacks.
- Provide screenshots of the cloned site running on both the local server and as viewed from a client device connected to the spoofed network.

USB Rubber Ducky lab

Overall difficulty: Easy/Medium

Suggested skills: Batch (.bat) scripting, basic Linux skills, general Android knowledge

This lab focuses on the USB Rubber Ducky, a device that emulates a keyboard while appearing as a standard flash drive. You'll get hands-on experience using the Rubber Ducky for task automation and explore its capabilities. Importantly, we'll also cover defensive strategies to protect yourself from potential keystroke injection attacks. By the end, you'll have a well-rounded understanding of this unique tool.

Setting up your USB Rubber Ducky

The USB Rubber Ducky is a straightforward device – it works without needing any extra software. Once plugged into a device, it launches Attack Mode HID[1] right away. To stop the attack and access the storage (where the *inject.bin* file lives), you just press a button on the device itself (see the picture below for reference). This design makes the USB Rubber Ducky incredibly easy to use for pre-programmed attack sequences. It's a powerful tool for ethical hackers and security researchers.

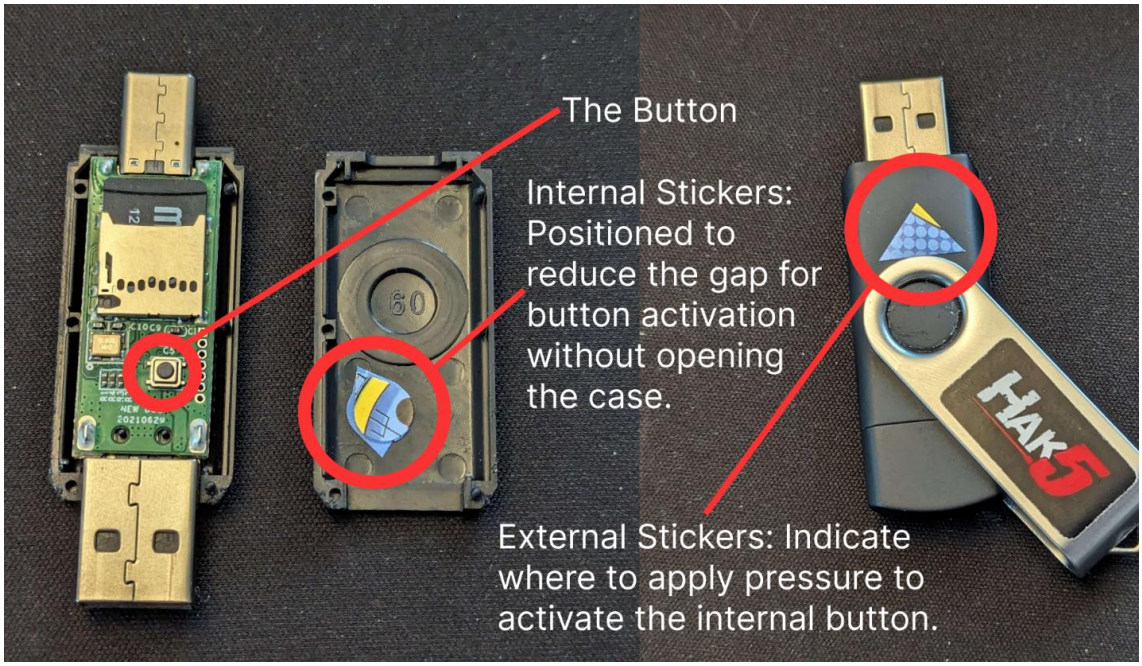

Figure 1 Internal button instructions

[1] Human Interface Device

Press the button to enable STORAGE mode, which allows you to access the root folder of the USB Rubber Ducky. For more information and guidance, I recommend referring to the official USB Rubber Ducky documentation. DuckyScript, the language used to control the USB Rubber Ducky, is straightforward to learn and you can write your scripts using Hak5's Payload Studio. Below are some of the most used commands of DuckyScript.

DuckyScript Cheat Sheet

- **REM** - Use this to add comments to your code.
- **DELAY** - Insert a delay in milliseconds.
- **STRING** - Type out text.
- **WINDOWS** or **GUI** - Press the Windows key.
- **SHIFT** - Simulate holding the Shift key.
- **ALT** - Simulate holding the Alt key.
- **CONTROL** or **CTRL** - Simulate holding the Ctrl key.
- **DOWN**, **UP**, **LEFT**, **RIGHT** - Press the arrow keys.
- **CAPSLOCK** - Toggle Caps Lock.
- **TAB** - Press the Tab key.

The USB Rubber Ducky offers two attack modes: HID mode, where it emulates a keyboard to inject keystrokes, and STORAGE mode, where it acts as a flash drive for file interactions. It's important to note that you can even use these modes together!

When starting a new payload, you can begin your script with ATTACKMODE [HID, STORAGE, or both]. If the ATTACKMODE is not specified, it will default to HID. Don't forget to include a brief DELAY after setting the attack mode; this gives your target computer time to properly recognize the device. After completing your DuckyScript, click *Generate* in Payload Studio, download the *inject.bin* file, and drop it into the root folder of your USB Rubber Ducky. Now your payload is ready to run! Try it with the code below:

```
DELAY 2000
GUI r
DELAY 800
STRING https://github.com/gjentig4
ENTER
DELAY 800
GUI r
DELAY 800
STRING notepad
ENTER
DELAY 1000
STRING I can type 9,000 keystrokes per minute (150 per second), but remember
to add delays after launching an app. Suggestions and comments are welcome on:
github.com/gjentig4. Don't run code you don't understand and be ethical!
ENTER
STRING Good luck!
```

If you'd like to learn more, there are plenty of resources available online. You'll find detailed information in their documentation, tons of helpful videos, and you can even ask the LLM of your choosing for additional insights.

Part I: Android Documents Fetcher (1p + 1p bonus)

Difficulty: Easy

Objective: Learn to write basic DuckyScript code and execute it on an Android device.

Setup:

1. **Create Your Payload:** Your goal is to design a DuckyScript payload that will grab the documents from an Android device. The device will be provided and unlocked for this exercise. There are multiple ways to tackle this, giving you a chance to experiment with the DuckyScript language.

2. **Prioritize Speed and Efficiency**: Your payload should execute within a minute, including all necessary delays. If you can optimize your script for retrieval in under 15 seconds, you will earn a bonus point.

> **Safety Note**: To protect personal data, use a designated test device and not your primary phone.
> **Android Tip**: The "GUI r" shortcut may provide direct access to the Files app on some Android devices.

Task: Document the execution of your payload on the Android device using a screen recording or video. If aiming for the bonus point with an execution time under 15 seconds, film the device externally with a stopwatch (physical, app-based, or on a website) clearly visible in the frame for accurate timekeeping like the figure below.

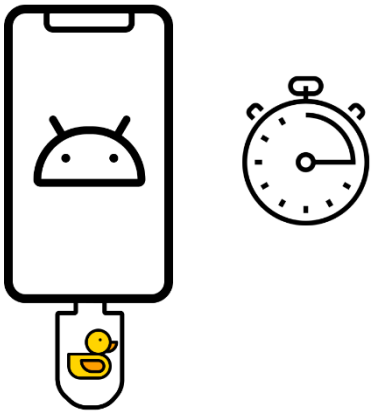

Figure 2 Instructions on filming the process for a bonus point.

Upload your recordings to YouTube (or another suitable video-sharing platform). Alternatively, convert them to a GIF for direct submission on OMA. Write a concise report (up to 400 words) explaining your approach, the DuckyScript code, and the results of your optimization efforts.

Part II: Reverse Shell Attack (2p + 1p bonus)

Difficulty: Medium

Objective: Establish a reverse shell connection to a target machine.

Setup:

This lab requires two machines:

- A device running Windows 11.
- A device running a Linux distribution of your choice.

This lab will explore establishing a reverse shell from the Linux machine to the Windows one with the help of USB Rubber Ducky. While you have flexibility in your methods, please ensure Windows Defender remains active on the target system.

> **Tip**: One approach utilizes the Metasploit framework. You'll need Metasploit framework and *Python3* installed on your host, an open port, and a *payload.exe* file created with *msfvenom*. The USB Rubber Ducky will trigger the download and execution of this payload on the target.
> **Note:** If the Windows 11 machine gets bricked, first attempt a system restore using the *Lab3Setup_restore_point*. If that method fails, reinstall Windows 11.

Task: Write a short report (of up to 500 words) describing the process:

- **Tools Used:** List and briefly explain the tools you employed (e.g., Metasploit framework, USB Rubber Ducky, etc.) and the role they played in this lab.
- **Host Machine Access**: Provide screenshots demonstrating the host machine successfully gaining access to the target machine.
- **Code and Commands**: Document all code written, and list all commands used on both the host and the target systems. Include clear explanations for each step.
- **Bonus point (1p)**: Take your skills further by establishing a persistent reverse shell connection. Describe and document the techniques used. *Tip:* This could also be done after first gaining initial access to the target machine.

- **Mitigation**: Write about how to defend against such attacks.

Part III: Document Extraction (4p + 1p bonus)

Difficulty: Medium/Hard

Objective: Fetch all *.txt*, *.docx*, and *.xlsx* documents from the target machine.

Setup:

This lab involves using a USB Rubber Ducky to extract *.txt, .docx*, and *.xlsx* documents from the target machine's Documents, Downloads, and Desktop folders, including all subfolders. The precise file names and locations within these directories are unknown; you will identify files by their extensions. Use only designated lab devices.

Preparation:

- Run the *LabSetup.bat* file on the target machine. This will generate the necessary folder structure and files which you must copy to the USB Rubber Ducky.
- Write a *.bat* file that searches for and extracts all targeted document types from the specified folders and their subfolders. Consider using command-line tools such as *xcopy* or *robocopy* for efficient searching and copying.
- Store your *.bat* file on the USB Rubber ducky and run it from there.

> **Note:** If the machine gets bricked, first attempt a system restore using the *Lab3Setup_restore_point*. If that method fails, reinstall Windows 11, and then execute *LabSetup.bat* to get everything set up again.

Execution:

- Deploy the USB Rubber Ducky on the target Windows 11 machine. Ensure that the script executes without alerting Windows Defender or other security measures.

Task: Write a report (of up to 600 words) to describe the process.

- Describe the steps taken from the initial script creation to the final execution of the payload on the target machine. Feel free to add older versions of your code and explain how it was optimized.
- **Code and Commands**: Provide the complete listing of the *.bat* file and the USB Rubber Ducky script. Include explanations of the code and commands, detailing how they contribute to achieving the lab's objective.
- **Outcome**: Present screenshots showing the successful extraction of documents. Highlight any challenges faced during the extraction process and how they were overcome.
- **Bonus Point (1p):** Modify your script so it does not fetch files that are 2MB or larger.
- **Mitigation**: Write about how to defend against such attacks.

O.MG Cable

While there were no labs developed for O.MG cables, if you enjoyed the lab, I encourage you to look into these cables that mimic ordinary USB-A, USB-C and Lightning cables but are actually designed for covert cyber-attacks. While USB Rubber Duckies are limited to pre-programmed payloads, O.MG Cables can be remotely triggered, granting attackers greater control over the timing of an attack. This disguise allows for easy, under-the-radar attacks. You can find different payloads made for O.MG here.

Their capabilities extend beyond simple keystroke injection. O.MG Cables can function as hardware keyloggers, secretly recording sensitive data. They have the potential to exfiltrate data from air-gapped environments and establish persistent backdoors into compromised systems.

It's important to understand the risks associated with O.MG Cables and be vigilant about their potential dangers. The best way to stay protected from them is to exercise caution by only using cables from reliable and trusted sources, regularly updating security software, and staying informed about the latest cyber security practices and threats.

To experiment with an O.MG cable in a lab, please ask your teacher to provide one.

Figure 3 O.MG Cables