UNIVERSITY of BRADFORD

FACULTY OF ENGINEERING AND INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

IoT HACKING: DEMONSTRATION OF CYBER SECURITY THROUGH PENETRATION TESTING ON Wi-Fi CAMERA and Raspberry Pi.

Student: Gogo Ebitimi Paul

UB: 22011871

Supervisor: Kit Qichun Zhang

A thesis submitted in part fulfilment of the degree of

MSc Cyber Security

ACKNOWLEDGEMENT

I would like to express my gratitude and appreciation to my supervisor Mr Kit Qichun Zhang my supervisor for his unwavering support and guidance throughout the course of this project. Additionally, I extend my heartfelt appreciation to my family and friends for all the unconditional support during this exceptionally demanding academic year.

ABSTRACT

The Internet of Things (IoT) has gained significant momentum across several domains, offering enhanced experiences in homes, logistics, healthcare, manufacturing, agriculture, smart cities, and other fields. For instance, IoT-enabled smart devices in agriculture automate real-time data collection, facilitating crop lifecycle monitoring and environmental condition tracking, such as water use, nutrient density, and fertilizer quantity, to increase production volumes, lower costs and enhance overall farming efficiency. Similarly, social networking and chatting applications have simplified human communication. IoT is ubiquitous, integrated securely through internet infrastructure, and interconnected on a large scale. However, there is a growing concern over the security of IoT. IoT devices collect data from every human movement, tracking their activity. This extensive data is stored on low-cost sensors and actuators. Unfortunately, the low production cost has not been favorable to high-level information security. IoT devices are susceptible to personal information or privacy leakage to unknown users, which can have severe consequences for individuals or business organisations, particularly in the hands of malicious users. The primary goal of this paper is to provide a comprehensive analysis and evaluation of security vulnerabilities in IoT systems. Both technical and non-technical aspects of these vulnerabilities will be explored, using ethical hacking methods and tools, provide practical security solutions that can be adopted based on the assessed risks, and ultimately seek to raise security awareness among users, manufacturers, and researchers.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

CHAPTER ONE

1.1 INTRODUCTION

Internet of things is very essential in today world and play integra part in our daily lives. The introduction of the Internet of Things has resulted in the ongoing global connectivity of individuals, items, sensors, and services. The Internet of Things' primary goal is to create a network infrastructure with software and communication protocols that are compatible with one another so that real and virtual sensors, PCs, smart devices, cars, refrigerators, microwaves, food processors, and medications can all be connected and integrated at any time and on any network [1]. The growing of internet of things has shaped so many industries such as health, finance, education, real estate, transportation etc. Networked cars, intelligent traffic systems, and sensors built into bridges and roads are examples of IoT technologies that are bringing us closer to the concept of "smart cities," which reduce traffic and energy use. By employing networked sensors to increase information availability along the production value chain, IoT technology presents the potential to revolutionise agriculture, industry, and the production and distribution of energy [2]. This technology is present in a broad range of networked systems, devices, and sensors. It makes use of developments in processing power, electronics miniaturisation, and network linkages to provide previously unattainable new capabilities. A lot of discussion has been channelled on the evolution of IoT which is cantered on the security of the devices powered by IoT. This has made researcher and industry experts to conduct research on the possible way to protect this devices connected to IoT from being attack.

According to [2], the widespread adoption of IoT devices holds great potential to revolutionise numerous facets of our lifestyle. Consumers are seeing a shift towards a "smart home" vision as a result of new Internet of Things (IoT) goods that provide increased security and energy efficiency, such as Internet-enabled appliances, home automation components, and energy management gadgets.

The primary concerns in an IoT context are security-related and include privacy, authorization, verification, access control, system setup, information storage, and management. IoT applications, which include embedded and smartphone technology, contribute to the creation of a digital world for worldwide connectivity that makes people's

lives easier by being perceptive, flexible, and sensitive to their requirements[1]. Security isn't certain, though. When a user's signal is lost or intercepted, their privacy could be violated and personal data could be disclosed. Conducting penetration testing on the IoT devices will gives industries or individuals the leads on how to protect their devices and block the loopholes that attacker can leverage on. The vulnerability assessment will ensured that the devices is protected from further exploitation.

1.2 Problem Statement

The growing use and dependence on Internet of Things (IoT) devices continue to increase in various aspects of our lives, the security of these devices has become a major concern. A recent study by Statista has projected that the number of IoT devices is set to more than double between 2020 and 2030, increasing from 9.7 billion to over 29 billion [3]. However, the widespread connectivity and data exchange of these devices has presented significant security challenges, making information security and privacy a pressing concern for individuals, businesses, and governments alike. While customers are drawn to the convenience and productivity gains provided by IoT devices, they often overlook the potential security or privacy risks associated with their use. IoT devices collect personal information to provide a personalized experience, but exploiting such devices leaves individuals and businesses vulnerable to severe consequences. A case in point is Wi-Fi cameras, which are commonly used for monitoring and surveillance. It is imperative that users and vendors pay closer attention to the serious implications that may arise if the security of these cameras is not taken seriously. Manufacturers often prioritise gaining a competitive edge, which means that security is often relegated to a lower priority. This costly trade-off provides malicious hackers with an opportunity-rich sandbox to develop exploits. Information security on these devices may be breached through system penetration exploits, which can escalate onto the network at large. It is crucial to raise cybersecurity awareness to reduce the number of successful cyberattacks. Therefore, increasing awareness across society can be beneficial.

1.3 Aim and Objectives

The project aims to conduct a comprehensive analysis of the security vulnerabilities present in some of the most common IoT devices of today namely the wireless CCTV security camera and Raspberry Pi (a minicomputer).

The objectives of this project are as follows:

1. leveraging ethical hacking techniques and tools to identify weaknesses in the IoT devices.
2. Establish a threat modelling process and provide Proof of Concept (PoC) that will be recorded and reported.
3. The research will also recommend appropriate security countermeasures to address and mitigate identified vulnerabilities found.

The goal of the study is to highlight how penetration testing can assist in identifying IoT security and privacy concerns.

1.4 Scope and Limitations

This paper focuses on hacking two of the most common IoT devices that are undeniably part of our daily lives since they are easily accessible, available, and affordable —exploiting the security vulnerability of the TP-link Tapo C200 camera and Raspberry Pi.

Limitations

However, not all security vulnerabilities will be covered, as limitations must be made to exclude some components and attack surfaces, such as physical and natural threats to IoT devices. Hence, the focus will be more comprehensive on their wireless communication ability to satisfy information assurance (IA). Also, the study shall conduct standard threat modelling of each IoT device mentioned above. If many vulnerabilities exist, the study shall use the DREAD risk rating model to score each discovered vulnerability.

1.5 Thesis Outline

The paper comprises six chapters, each dedicated to a specific aspect of the research. Chapter one serves as an introduction, presenting an overview of the thesis, its objectives, and its aim. Chapter two provides a review of existing research papers on IoT security vulnerabilities and other theoretical references of IoT penetration testing. Chapter three details the methodology implemented in this study. It explains the STRIDE and DREAD threat modelling techniques and their application to exploit unique vulnerabilities of the TP-Link Tapo c200 Wi-Fi camera and Raspberry Pi. Chapter four discusses the cyber-attacks and the Proof of Concept (PoC) exploit results with recommendations for mitigation strategies for

IoT security. Lastly, Chapter Five concludes the research paper and discusses recommendations for future work.

CHAPTER TWO

BACKGROUND STUDY AND LITERATURE REVIEW

2.1 IoT Definition

The Internet of Things (IoT) is a network that connects uniquely identifiable things to the Internet. The things have sensing/actuation and potential programmability capabilities [2].

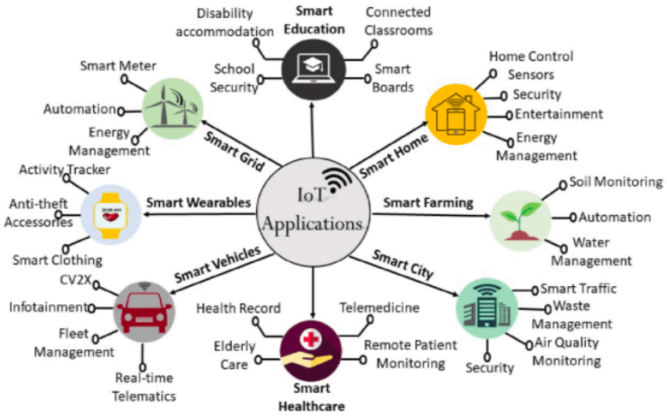

Figure 1: IoT applications.

Figure 1 captures an overview of various application domains of the Internet of Things (IoT) technology.

2.2 IoT Background

The Internet of Things (IoT) has transformed how we live since its inception in 1999 [2]. It has brought us to an interconnected world where even everyday objects like refrigerators and cars can be made "smart" or "intelligent" by connecting them to the internet [3]. Today, IoT encompasses a vast network of billions of physical devices spanning various application domains, including smart homes, smart grids, smart cities, smart healthcare, smart agriculture, smart transportation, and more [4].

Wireless sensor networks (WSN) form the foundation of IoT communication [5]. These sensors play a crucial role in collecting data from the physical world, which can then be transmitted to a local or cloud-based data center or another smart object via a gateway. Smart objects can also function as gateways, eliminating the need for intermediary devices. Once the data is received on the other end, multiple actions can be initiated, which can help

improve processes, reduce costs, enhance efficiency, and promote sustainability. Examples of IoT devices include home appliances, health-monitoring devices, wearables, unmanned aircraft systems (UAS), local and remote gateways, voice intelligent assistants like Google, Siri, Amazon Alexa, and low-power embedded devices like Amazon Echo, smart bulbs, smart plugs. These devices facilitate the fusion of the digital and physical world [3][4].

The IoT landscape has witnessed remarkable growth. According to Cisco's Annual Internet Report 2018-2023, the number of Internet-connected devices will exceed 30 billion by the end of 2023 [5]. The International Data Corporation (IDC) estimated that globally connected devices would reach 55.7 billion by 2025 [6]. This growth is expected to generate nearly 181 zettabytes of data in 2025, equivalent to a trillion gigabytes. Notably, 90% of the world's data was generated in the last two years alone [7]. However, the implementation process can be complicated, especially concerning interoperability within the IoT domain. Understanding the amount of data captured, processed, and created from the myriad of these connected devices periodically or in real-time gives the technology its value and has enabled it to become the world's fabric of information exchange today.

Despite its tremendous potential to transform the world and make it more connected, efficient, and sustainable, IoT is also a game-changer in the world of technology and beyond [8]. Any device that is exposed to the internet is vulnerable to cyber-attacks, and IoT devices are no exception. As more devices connect to the internet, the likelihood of attackers exploiting IoT device vulnerabilities increases. From attacks that steal data to those that may shut down critical infrastructures, the risk is significant [9]. In 2022, there were over 112 million cyber-attacks on IoT devices globally [10].

2.3 IoT Architecture

A systematic review of eight studies revealed that there are different models of IoT architecture, with some proposing three layers while others suggest four, five, six, or even seven layers. Despite the growing interest in IoT, there still needs to be more consensus on its architecture. The industry giants have published various IoT frameworks based on application requirements, network topology, protocols, business, and service models to complicate things further. These frameworks are designed to address the unique challenges posed by various industries and applications. Nonetheless, the most commonly used IoT

architecture follows a three-layer model. This model includes the perception, network, and application layers, which collect data from sensors, transmit data to the cloud, and process data, respectively [11, 12].

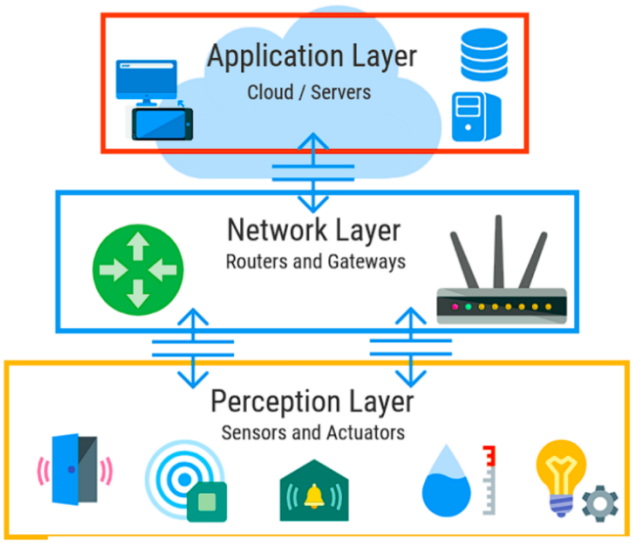

Figure 2: IoT Three-layered architecture

A. Perception layer

The Perception or physical layer consists of devices that collect information from their surroundings. These include sensors, actuators, and RFIDs.

B. Network layer

The Network layer forwards data collected by smart sensors to the application layer using communication channels and protocols for processing and transmission. Wireless protocols are commonly used in this layer as they are easily implemented. However, protocol selection varies based on speed, power consumption and hardware heterogeneity.

C. Application layer

The Application Layer provides a user interface for users to interact with the system.

2. 4 Common IoT Security Issues and Vulnerabilities

As the spread of IoT grows, many shortcomings have surfaced, and security is the biggest concern in IoT. First, it is important to understand the characteristics of what defines

security. Essential components of security in IoT are generally divided into three main categories [13]: confidentiality, integrity, and availability. Additional categories include authentication and non-repudiation [14].

1. **Confidentiality** implies information discretion from third-party disclosure [13, 14].
2. **Integrity** preserves information communicated remains unaltered or modified during transmission or delivery [13, 14].
3. **Authentication** determines the identity of source data is from the claimed identity or endpoint [13, 14].
4. **Non-repudiation** ensures action from an endpoint cannot later be denied.
5. **Availability** ensures information is available when needed [13, 14].

2.4.1 Expanded attack surface.

The rapid proliferation of IoT devices has introduced a multiple range of vulnerabilities and security challenges. their platforms such as cloud services (to communicate and store data), communication protocols (BLE, NFC and LPWAN) and even the systems to which they connected to pose great threat to information assurance (IA) attacks (e.g. router). Furthermore, tracking malicious flow would be overwhelming, especially when millions of devices communicate through the same channels. Reports by Check Point research, flagged a global increase in cyber-attacks targeting IoT devices to a weekly average of 54% with almost sixty (60)attacks per organisation [15], despite all its advantages, heterogeneity, flexibility, scalability, and ubiquitous connectivity which hampers its adaptation of improving efficiency today. It is without doubt that the technology is witnessing a notable growth in exploitation as it's becoming the next big attack surface for intruders [16]. Hackers now target businesses and critical infrastructures, as seen in recent attacks such as the Mirai, Tesla Model S remote hack, Saudi petrol chemical plant attack, Amazon ring hack, Dyn attack and Ukraine’s power grid witnessed deliberate and malicious attacks.

2.4.2 Lack of Complex Design

Vulnerabilities arising from the lack of encryption features are also a concern [17]. A substantial amount of research papers on IoT devices identified these devices as lightweight, and low-powered with limited memory, making it difficult to initiate

cryptographic functions and implement encryption protocols. They also process enormous amounts of data in real-time via the incorporation of Radio Frequency Identification (RFID)technology, WSNs, RSN sensing security in the devices, which has raised security concerns of unauthorised access as these are perceived as key technologies of IoT. RFID, a contactless identification resource of individuals and objects, enables an attacker to clone a tag's unique identifier and obtain sensitive information [18] by contacting a reader [19]. This example of poor security could serve as an entry point to exploit the victim's device and carry out a number of nefarious activities. The need for encryption in IoT devices is a significant challenge in ensuring information security.

A study by HP revealed 250 vulnerabilities in common consumer products connected to weak authentication protocols. The research included door locks, thermostats, webcams, TVs and home alarms that were easily exploited. The researchers could easily access most products, steal data and control devices remotely [20].

2.4.3 Lack of Standardisation

The absence of industry standardisation also poses a significant threat to the security of Internet of Things (IoT) devices and networks. Although IoT systems can autonomously collect and distribute data and information to another, the lack of standard communication and data formats protocols can cause interoperability issues and create vulnerabilities within the device or network, putting sensitive private data and information in danger. For example, if one device has a security vulnerability due to a lack of standards, it can be exploited, and the attacker can gain unauthorized access to other devices on the network. Similarly, if there are no standard data formats, data may become corrupted, lost, or manipulated by unauthorised parties. In essence, a lack of industry standardization can weaken the security of IoT devices and networks, making them more susceptible to attacks and data breaches.

2.5 Vulnerabilities Based on Owasp Top Ten

The Open Web Application Security Project (OWASP) presented a comprehensive vulnerabilities list of the top ten IoT devices and its ecosystem categorized below [21, 22, 23].

1. Weak, Guessable, or Hardcoded Passwords

IoT devices' web-based configuration, management, and authentication interfaces are often insecure. Weak default passwords and usernames are common, as are hardcoded credentials in firmware. Attackers can exploit these vulnerabilities to gain unauthorized access, leading to unauthorized changes to device configuration or sensitive information access [21, 22, 23].

2. **Insecure Network Services**

Using unneeded or insecure network protocols, services, configurations or outdated software can create vulnerabilities for attackers to exploit. This can result in the theft of sensitive data, unauthorized access, compromise of confidentiality, integrity/authenticity, or availability of information [21, 22, 23].

3. **Insecure Ecosystem Interfaces**

This vulnerability comes from insecure interfaces outside the device, such as web, backend API, cloud, or mobile interfaces. Attackers can exploit these interfaces to access sensitive data or control the device. Common issues include weak authentication/authorization, insufficient encryption, and input/output filtering. We will address these issues to provide secure solutions [21, 22, 23].

4. **Lack of Secure Update Mechanism**

IoT devices prioritize affordability, energy efficiency, and user-friendliness, but this can cause security to be overlooked during design. The lack of secure updating, firmware validation, secure delivery, and anti-rollback mechanisms leaves devices vulnerable to known vulnerabilities and exploits. Exploiting outdated firmware or software can compromise security, with severe consequences like financial loss or unauthorised access to sensitive information and disruption to critical systems, to name a few [21, 22, 23].

5. **Use of Insecure or Outdated Components**

The use of insecure or deprecated components/libraries in IoT devices is a growing concern in technology. Many IoT devices use third-party components that contain vulnerabilities, allowing attackers to compromise the device's security [21, 22, 23].

6. **Insufficient Privacy Protection**

Numerous IoT devices collect and store sensitive personal data but frequently lack sufficient privacy and data protection measures. This deficiency could result in the user's personal information stored on the device or ecosystem being exposed to unauthorized parties. Insecure or improper usage of such data or the lack of permission to access it could be disastrous and compromise critical information such as payment details sent over unencrypted channels [21, 22, 23].

7. **Insecure Data Transfer and Storage**

The lack of encryption or access control for sensitive data anywhere in the IoT ecosystem, whether at rest, in transit, or during processing, is a significant vulnerability. Attackers can intercept or manipulate data during transit or exploit weak storage mechanisms, compromising the security of the system. Therefore, it is crucial to implement robust encryption and access control measures throughout the IoT ecosystem to ensure the confidentiality of sensitive data and protect against unauthorized access or manipulation [21, 22, 23].

8. **Lack of Device Management**

Failure to effectively manage IoT devices can lead to network compromise. That may allow attackers to manipulate or control IoT devices remotely. Resulting in unauthorized access, firmware tampering, or device manipulation [21, 22, 23].

9. **Insecure Default Settings**

Many IoT devices are shipped with default settings and configurations that are often left unchanged by the manufacturer or users. These default settings may include generic usernames and passwords, open ports, and unencrypted communications, which can leave the device open to security risks. Attackers can exploit these vulnerabilities to gain access to the device, steal sensitive information, or carry out malicious activities [21, 22, 23].

10. **Lack of Physical Hardening**

The lack of physical hardening means that an IoT system still needs to implement necessary physical security measures. As a result, attackers may be able to gain access to sensitive information and tamper with the firmware, potentially giving them remote access to the device's local control [21, 22, 23].

2.6 Vulnerabilities Based on IoT Components

i. Physical Security Vulnerabilities

Physical security concerns related to IoT devices are often overlooked by vendors, primarily due to the historical lack of exploitation experienced by such products. However, these embedded devices are still vulnerable to a range of security challenges that arise from physical penetration, such as unauthorized access to sensitive components, memory devices, and processors. Intruders can modify existing code on the device software, leading to various security threats like denial of service (DoS) attacks, hardware trojans, eavesdropping, and physical tampering, which can all be accessed through an exposed interface. It is of utmost importance to address these challenges to ensure the effective realisation of the IoT.

ii. Security Protocol Vulnerabilities

Numerous security protocols are susceptible to attacks introduced in the protocol design, implementation and configuration stages [24]. Such vulnerabilities can provide external parties with a weak entrance to the network. For instance, in the context of the ZigBee protocol, an attacker can sniff the network and intercept the key exchange during device pairing. This is referred to as a man-in-the-middle attack (MiTM). Other examples of security protocol attacks include denial-of-service (DoS) attacks and malware attacks.

iii. Application Security Vulnerabilities

The Application Layer is more susceptible to security issues than the other two layers. This is because it includes various endpoints, such as web servers and mobile device applications (such as iPhone and Android). The Application Layer comprises the applications and software designed for IoT implementations, such as Smart Home applications. Threats and vulnerabilities can arise from any application code running on the host, which can compromise the host hardware directly or remotely. Notable attacks in this layer include

reverse engineering, malicious code injection, software-based modification, data integrity, and brute-force attacks.

iv. Wireless Reconnaissance and Mapping Attacks

Various wireless communication protocols such as ZigBee, ZWave, Bluetooth-LE, WiFi 801.11, among others, are utilized by numerous IoT devices. Unfortunately, hackers exploit this by deploying network scanning tools like Nmap, Nessus, Wireshark, Aircrack-ng, Nikto, and others to extract information about hosts, subnets, ports, and protocols. This can lead to network vulnerabilities, increasing the likelihood of a full-scale device attack [25].

2.7 Most Common Security Attacks in IoT

Attackers have various methods to gain access to an IT system. However, most cyber-attacks tend to follow similar strategies. These are the most prevalent IoT-based attacks:

1. Man-in-the-Middle Attacks

Man-in-the-middle (MitM) attack is a type of cyberattack where an intruder intercepts communications between two parties [26]. These attackers exploit weak protocols to insert themselves between entities in a communication channel and steal data [27]. MitM attacks are typically silent, making them difficult to detect [26]. The information obtained during an attack can be used for various purposes, including espionage, financial gain, or disruption. See

Table 1: Types of MITM attack

TYPES OF MAN-IN-THE-MIDDLE (MITM) ATTACKS	
	Email Hijacking
	Wi-Fi Eavesdropping
	DNS Spoofing
	Session Hijacking

	Secure Sockets Layer (SSL) Hijacking
	ARP Cache Poisoning
	IP Spoofing
	Stealing Browser Cookies

Effective MitM execution involves two distinct stages: interception and decryption [26]. During interception, the attacker intercepts client activity before it reaches its intended destination using techniques such as IP Spoofing, ARP Spoofing, Wi-Fi eavesdropping, DNS spoofing, or DNS poisoning. Decryption, on the other hand, occurs after any two-way SSL interception. Attackers can use techniques such as SSL hijacking, HTTPS spoofing, and SSL stripping [27]. In SSL hijacking, for instance, the attacker passes their authentication key to both the client and application during a TCP handshake, letting both parties know that the MitM controls the whole session. The most used communication channels for MitM attacks include HTTP, Bluetooth, NFC, RF, and Wi-Fi [28].

Man-in-the-middle (MitM) attacks have been responsible for some of the largest data breaches in recent history. One such example is the Equifax data breach in 2017, which exposed over 100 million customers' financial data to criminals over several months. Another example is the flaw in the banking apps of HSBC, NatWest, Co-op, Santander, and Allied Irish Bank, which allowed criminals to steal personal information and credentials, including passwords and pin codes. These incidents highlight the devastating impact MitM attacks can have on businesses and individuals alike [26].

2. Malware Attacks

Malware attack is one of the greatest of cyber-threats. The attack involves the attacker creating malicious software capable of causing harm or exploiting any programmable device, service, or network [29]. For over three decades, malware has been evolving and has found various attack vectors or method of infection such as email attachments, malvertising, fake software installations, infected USB drives, apps, phishing emails, and text

messages. The primary goal behind malware attacks is to extract sensitive data from victims that can be utilised for financial gains. Unfortunately, the prevalence of malware is rampant, with common types including viruses, ransomware, scareware, worms, spyware, Trojans, fileless malware, bots, rootkits and adware. Table 3 provides a comprehensive list of these types.

Table 2: Malware Attack

TYPES OF MALWARE ATTACKS	
	Ransomware
	Viruses
	Scareware
	Trojan
	Spyware
	Worms
	Botnets
	Rootkits

One real-life example recorded by [30] is the WannaCry ransomware attack of 2017 that caused damage to 230,000 computers globally, including one-third of all NHS hospitals in the UK. It exploited an operating system vulnerability that had a patch available before the attack, highlighting the issue of outdated systems. The attack resulted in approximately £92 million in damages and an estimated worldwide.

3. Distributed Denial of Service (DDoS) Attacks

One of the biggest security threats facing IoT is Distributed Denial of Service (DDoS) attacks. These attacks aim to make victim systems inaccessible to legitimate users by flooding them

with overwhelming traffic, leading to device slowdown or even crashes. Attackers form a network of bots, called a botnet, to carry out these attacks. Rather than hacking individual devices, a DDoS attack aims to infect a network of devices and create a massive army of bots that can be used to attack a large target. The attacker aims to add as many devices as possible to their botnet, which they control. When the attacker commands the botnet, it sends continuous requests and overloads the target [30].

The largest recorded distributed denial of service (DDoS) attack occurred when the Domain Name System (DNS) provider Dyn was targeted [30]. This attack caused most of the internet, including Twitter, Amazon, GitHub, and the New York Times to shut down. The perpetrators used a specific type of “botnet” malware, which originated from the infamous Mirai botnet attack. There have been other notable DDoS attacks, such as the AWS attack in 2020 and the GitHub attack in 2018.

4. Brute force Attacks

Brute force attack is a method hackers use to crack passwords, login credentials, and encryption keys in authentication attacks. It involves trying out all possible combinations through a trial-and-error approach to gain unauthorized access to a legitimate user's private account. Despite being an old technique, it is still highly effective and popular among hackers [31].

Table 3:Brute force attack type

TYPES OF BRUTE FORCE ATTACKS	
	Simple brute force attacks
	Dictionary attacks
	Hybrid brute force attacks
	Reverse brute force attacks
	Credential stuffing

One highly publicized brute force attack incident was the Dunkin' Donuts data breach, which resulted in the theft of 19,715 customer accounts and cost the company $650,000 in lawsuit settlement. Similarly, the Alibaba breach which compromised around 20.6 million accounts. However, the most notorious is the rockyou.txt wordlist data breach, in which over 32 million user passwords were exposed in plaintext. Today, the wordlist rockyou.txt has become a standard password-cracking tool [32]

5. Firmware Attacks

Firmware attacks are a serious threat to IoT devices, as they exploit vulnerabilities in the firmware that controls the hardware directly residing below the operating system[eh]. These attacks are particularly concerning since firmware stores sensitive information like credentials and encryption keys in memory. Attackers can obtain access to this sensitive data or control the device remotely, posing a significant risk to users' privacy and security. Firmware attacks are challenging to detect and can enable attackers to maintain access to a device for an extended period without the user's knowledge.

Two well-known examples of firmware attacks are the Robbinhood ransomware attack, which had root access to a victim's machine and encrypted all data, and the Thunderspy attack, which enabled attackers to read and copy all data on a device without leaving a trail, even when the device was shut down [33].

2.8 Most Common Countermeasures in IoT Security

As highlighted previously, security vulnerabilities within the Internet of Things (IoT) devices can result in a range of attacks, including the interception and theft of sensitive information, as well as the remote control of a device by malicious actors. To mitigate these risks, users and manufacturers must adhere to established security guidelines that promote best practices and safeguard against potential threats. In this regard, presented below are the most common countermeasures that can improve IoT security [34,35,36,37,38].

1. Device Authentication

Device authentication is crucial for device security and can be achieved through the Multi-factor Authentication (MFA) method recommended by NIST SP800-63 [35]. Hence, this guideline requires the use of an authenticator assurance level (AAL) framework to secure

any personal information available online. The MFA requires the user to provide their account credentials and a unique code sent to a secondary application, like email or text. For example, Two-factor authentication (2FA) provides an additional layer of security against brute-force attacks. It is an effective security measure to mitigate the risk of an attacker gaining unauthorized access to user accounts. Implementing 2FA enhances device and data security, protecting against potential cyber threats.

2. Data Encryption

The implementation of data encryption has emerged as a crucial strategy in mitigating the security threats associated with Internet of Things (IoT) devices and systems. Given the extensive data that these devices handle and their vulnerability to security breaches, encryption has assumed a multifaceted role in bolstering IoT security. Specifically, data encryption ensures the confidentiality of data by rendering it unreadable to unauthorized parties, maintains data integrity by detecting tampering, facilitates authentication through digital certificates and Public Key Infrastructure (PKI), controls access to authorized users, secures communication in public networks, enables end-to-end encryption for sensitive sectors such as healthcare and finance, ensures compliance with data protection regulations, manages encryption keys for robust security, protects firmware updates and boot processes, and reduces the attack surface by impeding unauthorized access. In implementing data encryption, it is crucial to balance security and resource constraints, necessitating regular security assessments and updates that can adapt to evolving IoT threats and vulnerabilities [36].

3. Firewall

Firewall is a fundamental mitigation strategy for safeguarding computer networks and systems against cyber threats [36]. Acting as a barrier between a trusted internal network and untrusted external networks (such as the Internet), they prevent unauthorized access and filter incoming and outgoing network traffic based on a predetermined set of security rules. Firewalls offer a range of sophisticated functions as a mitigation measure, including access control, intrusion prevention, application layer filtering, network segmentation, content filtering, virtual private networks (VPNs), stateful inspection, logging and monitoring, policy enforcement, scalability, and performance. These features are

instrumental in enabling organizations to enforce security policies, comply with regulatory requirements, and protect against various cyberattacks, including malware infections, port scanning, denial-of-service attacks, and intrusion attempts [37,38].

4. Penetration Testing

IoT security heavily relies on penetration testing to ensure the safety of internet-connected devices. It is crucial for manufacturers to allow their devices to undergo rigorous testing before releasing them to the market. A team of cybersecurity analysts conduct meticulous examinations to identify any potential weaknesses or vulnerabilities. Where the primary goal of these tests is to assess the impact of possible exploits and to uncover flaws that malicious actors could exploit. If any issues are found during the testing process, the device security team, known as the "Blue" team, is promptly notified [34]. They then implement necessary patches while the devices are still in production. This comprehensive testing approach is essential in eliminating vulnerabilities, preventing them from reaching customers and enhancing the overall security of IoT devices. By subjecting devices to rigorous testing, manufacturers can install trust and confidence in their customers, knowing that the products they purchase have undergone thorough testing to ensure their safety and security.

5. Network Segmentation

Network segmentation involves the creation of segregated network segments or Virtual Local Area Networks (VLANs) specifically for IoT devices. This isolation effectively prevents unauthorized access to critical resources in the core network. Even in the event of an IoT device being compromised, attackers are limited to the segmented network, thereby restricting their ability to move laterally within the network and causing more severe damage/consequences. By isolating IoT devices, this can significantly reduce the attack surface, limiting the number of entry points and opportunities for attackers to exploit vulnerabilities. An example of this is the use of smart plugs or smart lightbulbs, which typically do not require sensitive information to operate. Therefore, if there is an intrusion, their individual network can be terminated as a final measure without inflicting further damage to the primary network or the devices linked to it [37,38].

6. Monitoring and Logging

By implementing a monitoring and logging strategy, IoT environments can proactively identify and address security threats, safeguarding the integrity and security of IoT devices and data. This mitigation strategy entails several key components. Real-time monitoring through continuous log data analysis is essential in detecting anomalies and irregular device behavior. Effective alerting mechanisms that prioritize alert severity for rapid response to security breaches. The integration of machine learning and artificial intelligence (AI) can enhance log analysis, enabling the identification of complex and evolving threats. Behavioral analytics establish a baseline for typical IoT device behavior, triggering alerts when deviations occur. Access control measures restrict log access to authorized personnel, safeguarding logs against unauthorized manipulation. Secure transmission of log data through encryption and the use of data integrity checks and timestamping ensures the accuracy and authenticity of log entries. An incident response plan complete with roles and responsibilities should be developed, and regular reviews and updates conducted to adapt to evolving threats. User education to prompt reporting of suspicious activities. Third-party security information and event management (SIEM) systems tailored for IoT environments can provide advanced monitoring and logging capabilities. Monitoring and logging help identify potential threats and vulnerabilities and enables timely incident response.

2.8.1 Other Common Mitigation Strategies

There are a variety of additional methods that both manufacturers and end-users can employ to ensure the security of their devices and the IoT network, in addition to the practices. These measures can provide a solid foundation for improving the security of IoT devices and mitigating the potential risks that come with them [34,35,36,37,38]. This includes.

- Device Identity Management
- Secure Boot
- Firmware and Software updates
- Access Control Lists (ACLs)

- Physical device security

2.9 Related Works

In this section, the related studies to IoT application security are highlighted. One study [39] provided an overview of IoT protocols and vulnerabilities through security penetration testing on the Belkin WeMo Smart plug. The study categorized IoT vulnerabilities based on the OWSAP Top 10 IoT vulnerabilities and discussed security challenges based on the IoT architecture. However, mitigation strategies for IoT security were not outlined in the study. Other studies [40, 41, 42, and 44] provided comprehensive analyses of security issues, and reference 43 covered cybersecurity and IoT domains, including penetration testing, common attacks, and communication protocols. The authors of reference 42 covered IoT device architecture and penetration testing methodology and provided a taxonomy review of the three major layers of importance in the IoT system framework: application levels, network, and perception. Meanwhile, references 40 and 44 discussed some mitigation strategies. However, each of these studies has limitations in coverage to address these limitations. This review presents a general approach to IoT security by covering the fundamentals of IoT, including standard architecture, vulnerabilities, protocols, attacks, and mitigation strategies for IoT security challenges.

CHAPTER THREE

DESIGN AND METHODOLOGY

3.1 Threat Methodologies

The threat modelling technique is used to gain a thorough understanding of the attack process before conducting penetration testing. This technique breaks down the involved processes, providing a more nuanced understanding of the various attack perspectives.

Penetration testers use threat modelling to analyze the technical features of a device and identify potential security vulnerabilities. The attack surface refers to the ways in which a device can be compromised. The more attack surfaces a device has, the higher the likelihood of it being compromised. The six most common threat modelling methodologies used to assess threats to an IT asset are STRIDE, PASTA, VAST, TRIKE, DREAD, and OCTAVE. This study shall adopt the STRIDE and DREAD approaches based on the objectives of the paper and project timeframe.

3.2 Network Design

3. 2.1 Environmental Setup Camera

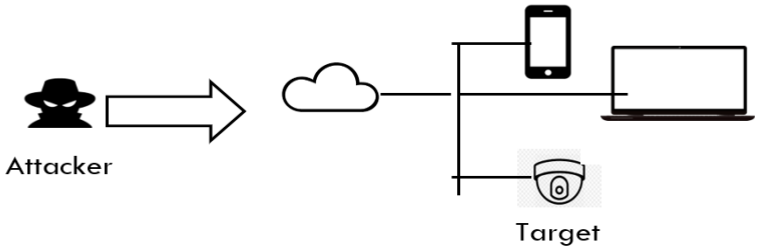

Figure 3: Camera Attack Setup

3.2.2 Environmental Setup Raspberry Pi

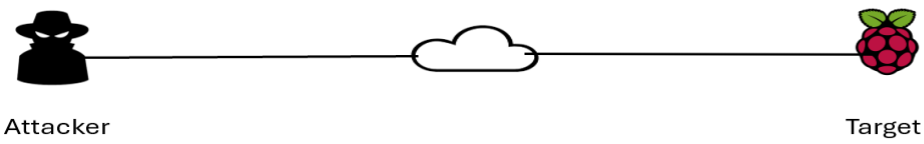

Figure 4: Raspberry Attack Setup

3.4 Security Analysis

1. STRIDE is a mnemonic for a set of threats- Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service (DoS), and Elevation of Privilege as described in the table below:

Table 4:Stride Model

Threat	Security property violated	Threat definition
Spoofing	Authentication	Attacker impersonating something or someone else.
Tampering	Integrity	Attacker modifying data or code
Repudiation	Non-repudiation	Attacker claiming to have not performed an action.
Information Disclosure	Confidentiality	Attacker exposing information to someone not authorized to see it
Denial of Service	Availability	Attackers deny or degrade service to users
Elevation of Privilege	Authorization	Attackers gain administrative capabilities without proper authorization

2. Identifying the asset

The first step towards creating a threat model is to identify all assets.

Table 5: Asset Identification

Asset	**Description**
Camera	The camera model TP-Link Tapo C200 for iOS and android CCTV camera. With supporting technologies and protocols such as Bluetooth speaker, Wi-Fi, HTTPS, RTSP, UDP, built in micro-SD card and motion detection.
Mobile Application	TP-Link Tapo (Smart Home app) is a mobile application that serves as the user interface providing real-time streaming and allowing quick and easy control of all TP-Link smart life

	products.
Wireless communication	The camera and the Cloud sever communicate using Wi-Fi 802.11/b/g/n 2.4GHz
Cloud services	Cloud services are used to upload or store the camera's video recordings. Port 8800 supports this service.
Firmware	Firmware version 1.3.5. This controls various configurations and can be upgraded.
Smartphone	Samsung Galaxy s6 (Smartphone) an android OS mobile device which will be used to control the TP-Link Tapo C200 smart camera.
Raspberry Pi	A minicomputer

3.4.1 Threat Modelling of Wireless Security Camera

i. Identifying threats

Threat #1

Table 6:Unauthorized access to user's account

Threat Description	Unauthorised access to user's account
Threat Target	Mobile/Web application
Attack Method	Brute-force techniques
Countermeasure	Admin should set up unique and complex password and where possible enable account lockout.

Threat #2

Table 7:Access to video recording or picture files storage

Threat Description	Access to video recordings or picture files storage.
Threat Target	Mobile application/Web application
Attack Method	An attacker could access files stored such as video recordings.

Countermeasure	Admin should enable two- factor authentication that way attacker would not be able to access without the user's permission

Threat #3

Table 8:Packet Sniffing

Threat Description	Packet Sniffing
Threat Target	Network
Attack Method	An attack can use any sniffing tool such as Wireshark to intercept packet or eavesdrop on communication channel.
Countermeasure	Admin should set up a virtual private network (VPN), as this will ensure encrypted wireless communication.

Threat #4

Table 9:Malicious firmware

Threat Description	Malicious firmware
Threat Target	Tapo c200 Camera Firmware
Attack Method	An attacker can create a malicious firmware and run it onto the camera's built-in micro-SD memory card.
Countermeasure	Physical access to the camera should be restricted likewise downloads and upgrades should only be initiated from authenticated sources.

Threat #5

Table 10:An attacker pretends to be a router to intercept traffic.

Threat Description	An attacker pretends to be a router to intercept traffic.
Threat Target	Samsung galaxy s6

Attack Method	Man-in-the-Middle technique
Countermeasure	Encrytpt packet and traffic visibility such as tunnelling or VPN.

Threat #6

Table 11:An attacker may attempt to find open ports

Threat Description	An attacker may attempt to find open ports
Threat Target	Tapo c200 Camera
Attack Method	Port Scanning using Nmap
Countermeasure	Admin can set up a firewall

3.4.2 Threat Modelling of Raspberry PI

Threat #7

Table 12:Backdoor attack

Threat Description	Backdoor attack
Threat Target	Raspberry PI
Attack Method	Backdoor attack using Metasploit framework and payloads
Countermeasure	Firewall and/or IDS, Honeypots,

Threat #8

Table 13:MITM attack

Threat Description	MiTM attack
Threat Target	Raspberry PI
Attack Method	MITM attack using Ettercap or Wireshark
Countermeasure	Admin can Implement a VPN router on a Raspberry Pi to encrypt end-to-end connections, preventing data interception by attackers.

DREAD threat Modelling.

The DREAD model is a quantitative tool that measures the severity of cyber threats using a rating system that assigns numerical values to different risk categories. The model consists of five categories:

Damage Potential: How much damage could the attack cause?

1. **Reproducibility**: How easy it is to replicate an attack?
2. **Exploitability**: How easy is it to attack?
3. **Affected Users**: How many users would be affected?
4. **Discoverability**: how easy it is to discover the vulnerability?

To evaluate and rank the seriousness of potential threats discovered by the STRIDE model, the DREAD rating system will be used. This system assigns a score ranging from 0 and 10 in each of the five categories mentioned above. The final rating, calculated as the average of these category ratings, indicates the overall severity of the risk. See Appendix I

The overall threat rating is determined by adding up the scores from the five key areas. The risk severity for a threat is divided into four categories:

Table 14:DREAD risk rating.

Risk Rating	**DREAD score**
Critical	40-50
High	25-39
Medium	11-24
Low	1-10

3.4.3 Threat Rating of Security Camera

Threat #1

Table 15:Unauthorized access to user's account: rating score

Unauthorised access to user's account	**SCORE**
Damage Potential	10
Reproducibility	5
Exploitability	5
Affected Users	8

Discoverability	0
OVERALL RISK RATING: HIGH	27

Threat #2

Table 16:Access to video recordings or picture files storage.

Access to video recordings or picture files storage.	**SCORE**
Damage Potential	0
Reproducibility	0
Exploitability	6
Affected Users	6
Discoverability	5
OVERALL RISK RATING: MEDIUM	17

Threat #3

Table 17:Packet sniffing rating

Packet Sniffing	**SCORE**
Damage Potential	0
Reproducibility	10
Exploitability	5
Affected Users	0
Discoverability	8
OVERALL RISK RATING: MEDIUM	23

Threat #4

Table 18:Malicious Firmware rating

Malicious firmware	**SCORE**
Damage Potential	10
Reproducibility	10
Exploitability	5
Affected Users	10
Discoverability	0
OVERALL RISK RATING: HIGH	35

Threat #5

Table 19:Traffic Interception rating

An attacker pretends to be a router to intercept traffic.	**SCORE**
Damage Potential	0
Reproducibility	5
Exploitability	5
Affected Users	6
Discoverability	5
OVERALL RISK RATING: MEDIUM	21

Threat #6

Table 20:Scan open ports rating

An attacker may attempt to find open ports	**SCORE**
Damage Potential	5
Reproducibility	7.5
Exploitability	5

Affected Users	0
Discoverability	5
OVERALL RISK RATING: MEDIUM	22.5

3.4.4 Threat Rating of Raspberry PI

Threat #7

Table 21:MITM attack

An attacker may attempt to find open ports	**SCORE**
Damage Potential	6
Reproducibility	5
Exploitability	5
Affected Users	10
Discoverability	0
OVERALL RISK RATING: HIGH	26

Threat #8

Table 22:Backdoor attack

An attacker may attempt to find open ports	**SCORE**
Damage Potential	10
Reproducibility	5
Exploitability	5
Affected Users	10
Discoverability	0
OVERALL RISK RATING: HIGH	30

CHAPTER FOUR

PENETRATION TESTING, RESULTS AND DISCUSSION

The purpose of this section is to demonstrate the penetration exercise executing on the IOT device Wifi Camera as well as the Raspberry Pi.

4. 1 Wifi Camera For Pen Testing

A penetration test involves identifying and exploiting vulnerabilities within a device or software. This segment illustrates the process of uncovering vulnerabilities in the TP-link C200 wifi camera and how exploitation could result in sensitive information being stolen or device functionality being disrupted. Cameras are typically installed for security purposes, but a compromise could prevent them from recording video, allowing thieves to go undetected.

4.2 Tapo C200 Camera

Tapo is a brand name under TP-Link, a world renoved networking company. Tapo devices includes cameras, are designed for smart home appliances, officess and the community to security purposes. For this research, Tapo C200 Camera was picked to demostrate how IOT device can be hacked if the vulnerability are not mitigated.

Figure 5: Tapo Camera

4.2.1 Setup of Tapo C2oo Wifi Camera

To make sure the camera and mobile application worked as intended and to get to know the device, the camera was installed in line with the camera user handbook. To begin customising the Tapo C200, the TP-Link Tapo Application has to be downloaded from Google Play or the App Store. Since the test environment was an Android device, the application was downloaded via the Google Play Store. When the application was opened, it was necessary to configure a TP-Link ID, which was accomplished by creating an account using an email address. After creating and verifying the account with the email, the Tapo C200 model may be selected in the user interface to add the device. The camera and smartphone need to be set up to work together.

4.2.2 Reconnaissance

Reconnaissance is a process of finding information about a potential target to carry out an attack. This process can be carried in two different method, either through active and passive method. For this execise, a passive method was deployed to gather information about the target which is Tapo C200 camera.

Tapo C200 Camera is security camera connected via wireless network. The camera can be stream via Real time transmission protocol (RTSP). The camera also has pan and tilt capabilities, night vision, and live video streaming. It has data sharing capabilities with many users, privacy mode to protect user privacy, and an optional 128 GB external microSD storage slot that is not packaged. The LED light on the camera may blink or change colour depending on the settings. The Tapo app, available on Google Play and the App Store, is compatible with the camera, which communicates and stores data via the TP-Link Cloud server.

The application provides features like playback, motion detection alert setting, pan and tilting of the camera, and talking or voice calling. The network makes use of 802.11 b/g/n at 2.4 GHz, WPA/WPA2-PSK, SSL, TLS, and 128 bit AES encryption. The system has regulatory certifications from the FCC, IC, CE, and NCC, and information about the camera sensor's range, resolution, and night vision is included. For it to work, iOS 10 or Android 5.0 are needed. The three hardware revisions of the camera are V1, V2, and V3. Using RTSP2, a camera account can be set up in the mobile application to allow camera viewing via a PC. The video feed can be seen via one of two different URLs.

4.2.3 Finding Vulnerability

Vulnerability are weakness that exist in a software of device that can be exploited by attackers to gain access illegally. To find vulnerability in the tapo c200 camera, a vulnerability database was used to search for potential loopholes that associated with the camera.

The search through https://www.nvd.nist.gov shows the vulnerability associated with Tapo C200. According to the result as shown in figure…, The vulnerability ID: CVE-2023-27126, “the AES Key-IV pair used by the TP-Link TAPO C200 camera V3 on firmware version 1.1.22 Build 220725 is reused across all cameras. An attacker with physical access to the camera is able to extract and decrypt sensitive data containing the wifi password and the TP-Link account credential of the victim”. The second vulnerability is CVE-2021-4045 which has to do with unauthenticated RCE vulnerability, present in the uhttpd binary running by default as root. The exploitation of this weakness can allow attacker to take full control of the camera. The vulnerability is rated high which is a serious concern that needs to be addressed.

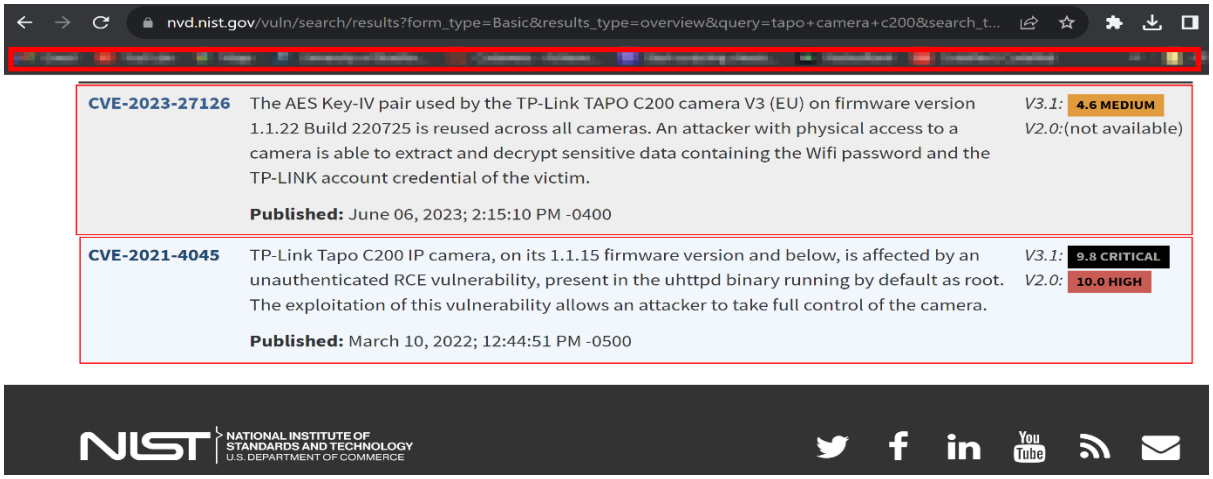

Figure 6: Vulnerability

4.2.4 Scanning

Scanning is another form of gathering information about a potential target. Having done reconnaissance, the next stage is to carry out scanning to find what IP address the TAPO C200 is connected within the wireless network. To find possible targets such as Tapo C200 Camera, and other devices because the attacker has no knowledge of their existence, the first step is to run a scanning on the local network. Network Mapper (Nmap) is used to scan the whole network within the region. As shown in Fig… sudo nmap -sP 192.168.4.0/24 was the command used to find the possible live host in the network. From the result shows, a total four host were discovered to be alive.

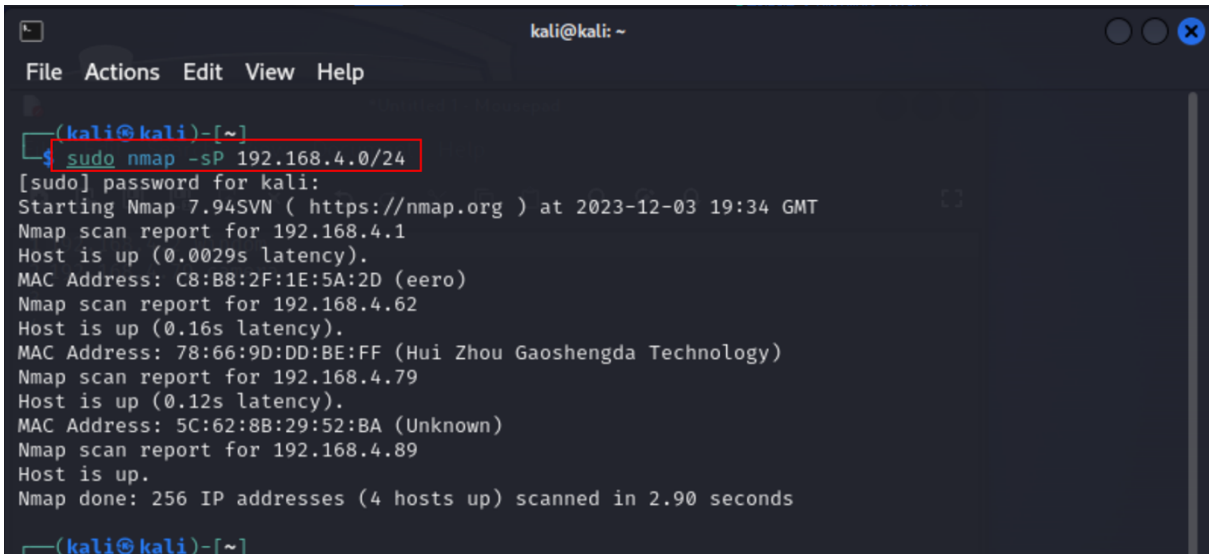

Figure 7: Nmap Scanning

To intensive the enumeration further, scanning were done for all the four devices to ascerntain the IP that belongs the to TAPO C200 camera. As seen in figure .. the result returned that IP address 192.168.4.79 is said to belonged to TAPO C200 camera. The scanning reveal ports currently running on the IP which includes RTSP protocol.

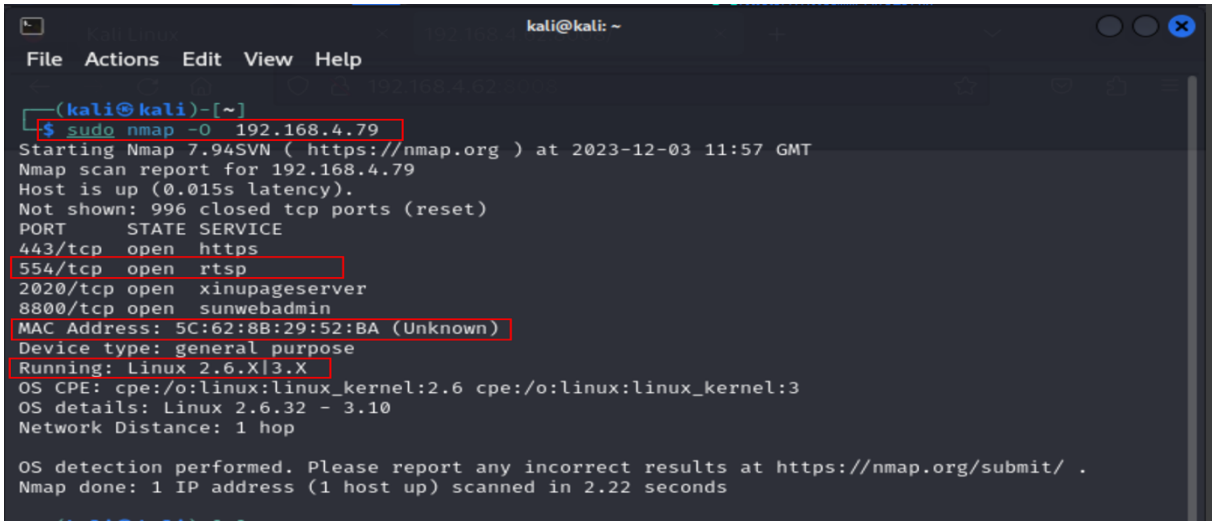

Figure 8: RTSP Port

4.3 Exploitation

Exploitation can be regards as penetration. This process involves exploited the vulnerability to gain access to the TAPO C200 Camera. In exploiting the TAPO C200 Camera, dictionary attack method was used to hack the username and password. The penetration will give attacker access to stream the camera remotely.

4.3.1 Dictionary Attack

In conducting the attack, hydra tool was used to crack the username and password. Having discovered that, if the login and password have been hacked, an attacker may be able to watch the WiFi camera remotely through port 554, which is used for real-time streaming protocol (RTSP). This section showed how to use Hydra to hack a login and password. Since Hydra is using a dictionary attack, it will need a list of usernames and passwords. Despite lacking the device login credentials, the attacker is attempting to obtain access by whatever means feasible.

The effectiveness of the hydra-based attack on the Tapo C200 camera is seen in Figure. On the device, Hydra managed to locate a username and password. Following a carryout attack on the camera, Hydra found admin1 and password123 as usernames and passwords, respectively. The command used was to tell Hydra to look at the file name that was submitted, username.txt to see if it matches the device's username, and password.txt to see if the device has a potential password.

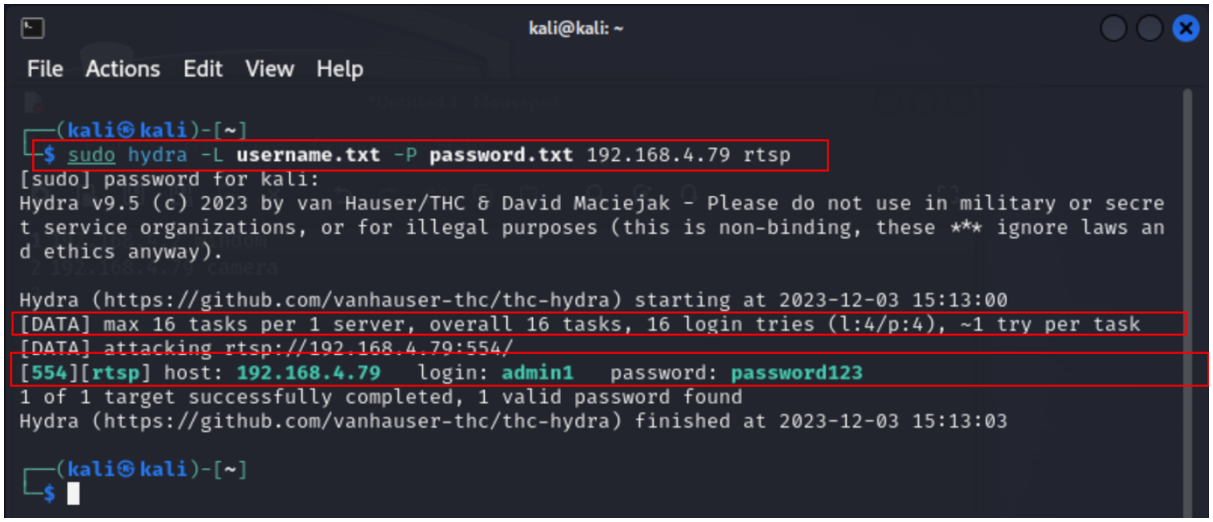

Figure 9: Dictionary Attack

The next step is to demonstrate how the attacker gained access to the target's login credentials and streamed the camera. The attacker used a VLC player to view and stream the video.

The figure shows an attacker trying to get access to the page using a false password, which causes a prompt to appear asking for a correct password before the attack.

URL: rtsp://username:password@host:port number/stream1

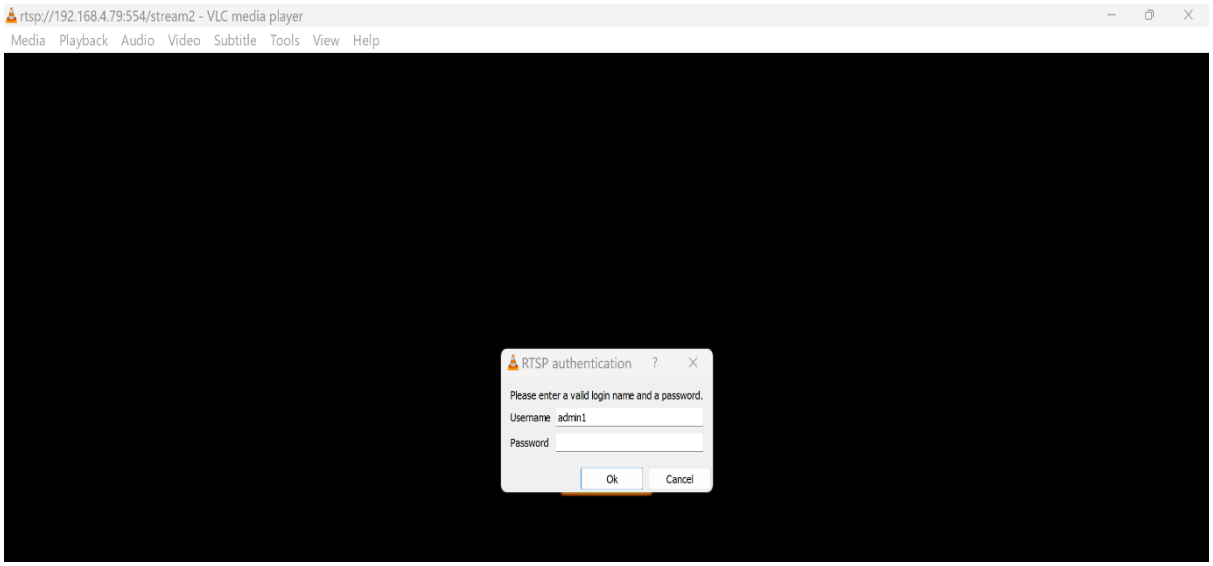

Figure 10: RTSP Login by Attacker

Upon entering the correct login credentials, the attacker was able to stream the camera live as shown in Figure. The figure displaying the image of from the camera as been stream remotely by the attacker. It's show the date and the time of the view, 2023-12-03 17:39:22.

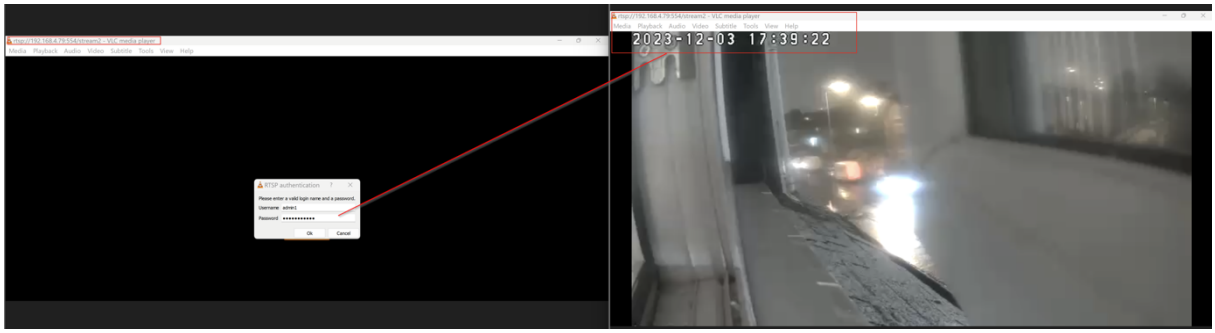

Figure 11: Camera image after successful login by attacker

Security Impact of Dictionary Attack on Camera

A dictionary attack on cameras could jeopardized the surveillance system's availability, confidentiality, and integrity, which could have serious security repercussions. These are some significant security implications.

1. Confidentiality: An attacker could breach the privacy and confidentiality of the surveillance data by seeing live or recorded footage if they are successful in performing a dictionary attack and get access to the camera system as earlier shown.
2. Integrity: If an attacker gains access, they might alter or remove recorded video, which would mean important evidence is lost. This may compromise the surveillance system's integrity and impair its capacity to deliver accurate and trustworthy data.
3. Availability: A successful dictionary attack might cause unauthorised adjustments to camera settings, which would interfere with the surveillance system's capacity to function. This can entail turning off cameras, changing setups, or bringing about other hiccups in service.
4. Credential: If the same username and password combinations are used for several services or devices, the impact of a successful attack on one camera may be increased if it results in unauthorised access to other accounts or systems.
5. Reputation Damage: A security breech could harm the standing of the company or person in charge of the camera system. If privacy laws are broken, users might stop trusting the security safeguards put in place, and the company might end up in legal hot water.

4.3.2 Mitigation Techniques

Mitigation Techniques is a measure put in place to protect device from being hacked. This measure will strengthen the security posture of the camera. The following security measures is needed to protect against dictionary attack.

1. Intrusion Detection and Prevention System (IDPS): Deploy IDPS to track network activity and identify trends that could point to dictionary attacks. This measure would ensure victims is notify of any potential attack on the camera.

2. Logging and Monitoring: Implement reliable logging systems to monitor login attempts and identify any unusual trends.
3. Regular software update: Ensure that the firmware and software on the TAPO C200 Camera is constantly updated.
4. Network Segmentation: It's important to separate CCTV camera from others network to contain impact of attack. Implementation of firewalls and access controls to restrict access to camera by unauthorised accessed is very important.
5. Strong password policies: Enforce use of strong and complex password and regularly changing of password to avoid easily guessable information.

4. 4 Deauthentication Attack

A deauthentication attack entails cutting off a device's connectivity to a wireless access point, which interferes with the device's functionality and renders it unusable for its intended usage.

The attack was carried out using three distinct tools: aireplay, airodump-ng, and airomon-ng. With the use of a wireless adapter called the ALFA Network (IEEE 802.11b/g/n) Long-range USB adapter—model number AWUS036NHA—the assault was made possible. Furthermore, STRONG Inc., a wireless router, was instrumental in making the attack possible.

Device Information

Product model	4G+Router1200
Hardware version	V2.0
Software version	V2.0.0
SN	121703082200223
IMEI	861572050032909
IMSI	234202502220355
PLMN	23420
Running time	1Day15Hour10Minute

I. Airmon-ng

A command-line tool for wireless network auditing, airmon-ng is a member of the Aircrack-ng suite of tools. A popular tool for assessing Wi-Fi network security is Aircrack-ng. Airmon-ng is used to enter a monitoring status for the wlan0 interface. Moreover, it can be used to terminate ongoing processes on the interface. Enabling monitor mode ensures that you can use a wireless adapter with VirtualBox to track available wireless networks. Figure...

demonstrates how to use airmon-ng to switch wlan0 from managed mode to monitor mode.

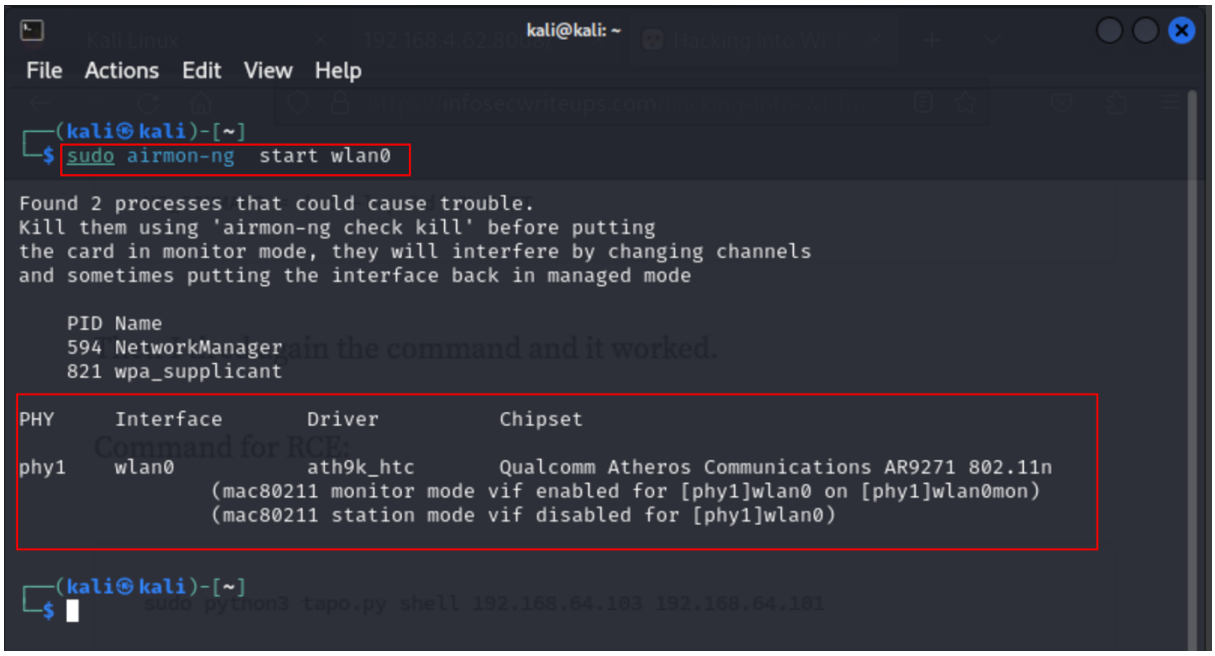

Figure 12: Starting Airmon

II. Airodump-ng

Another effective tool in the Aircrack-ng toolkit for packet capturing and wireless network monitoring is Airodump-ng. It is very helpful for collecting data on the accessible networks, analysing Wi-Fi networks, and capturing data packets.

During packet capture, raw 802.11 frames are recorded using Airodump-ng. It's particularly useful for collecting WEP IVs (Initialization Vector) or WPA handshakes to use with aircrack-ng. If a GPS receiver is connected to the PC, airodump-ng can be used to record the GPS coordinates of the access points that are found. Furthermore, a variety of files are generated by airodump-ng that provide details about every client and access point it has encountered. Everyone can script with these files or utilise them to create new tools. Figure... shows how to capture the wireless network using airodump-ng.

Command: **sudo airodump-ng wlan0mon** display the list of available wireless access point. The highlighted access point is the target network.

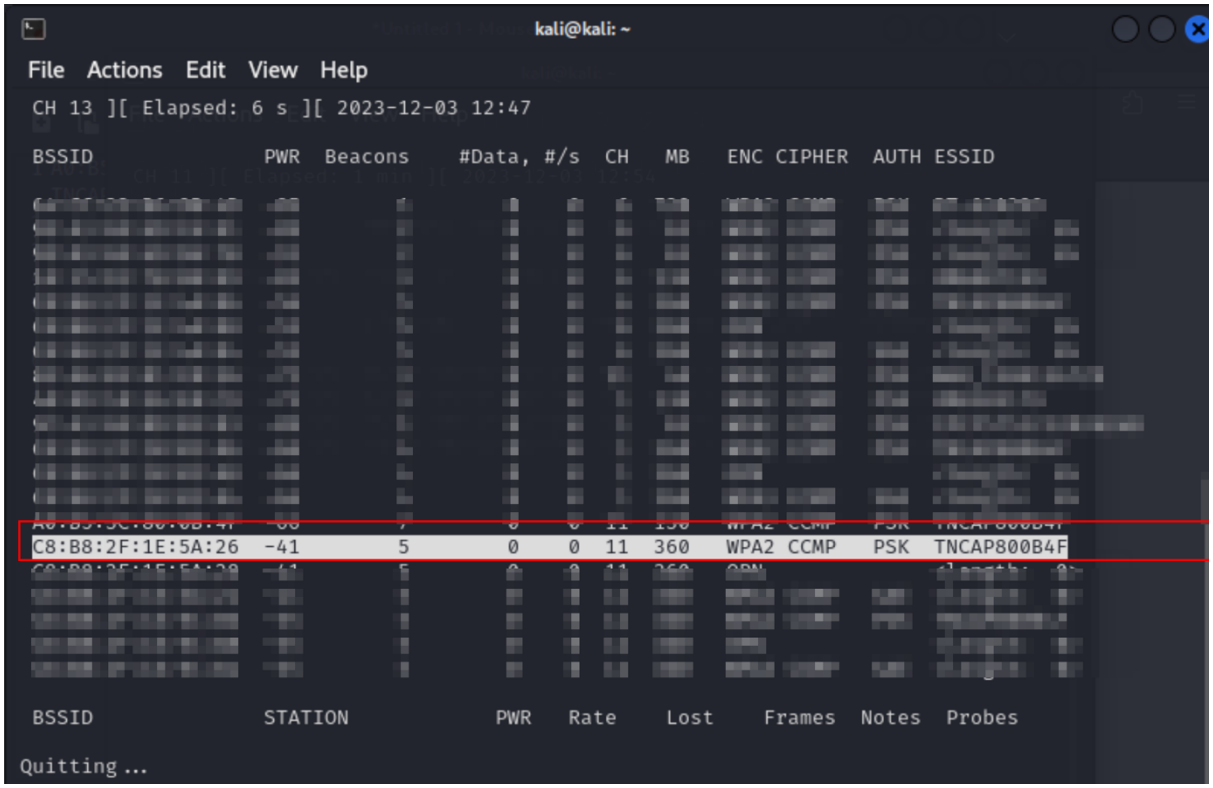

Figure 13: Wireless access point discovery

To see the list of available devices connected to the access point, “airodump-ng” command was used. “sudo airodump-ng --channel 11 --bssid C8:B8:2F:1E:5A:26 wlan0mon” revealed three devices were connected to the wireless access point. The Tapo camera is one of these devices to be disconnected from the network. The Tapo Camera was identified with MAC address "5C:62:8B:29:52:BA". The figure depicts the list of available devices connected to the access point. The highlighted device is the target that would be disconnected from the access point.

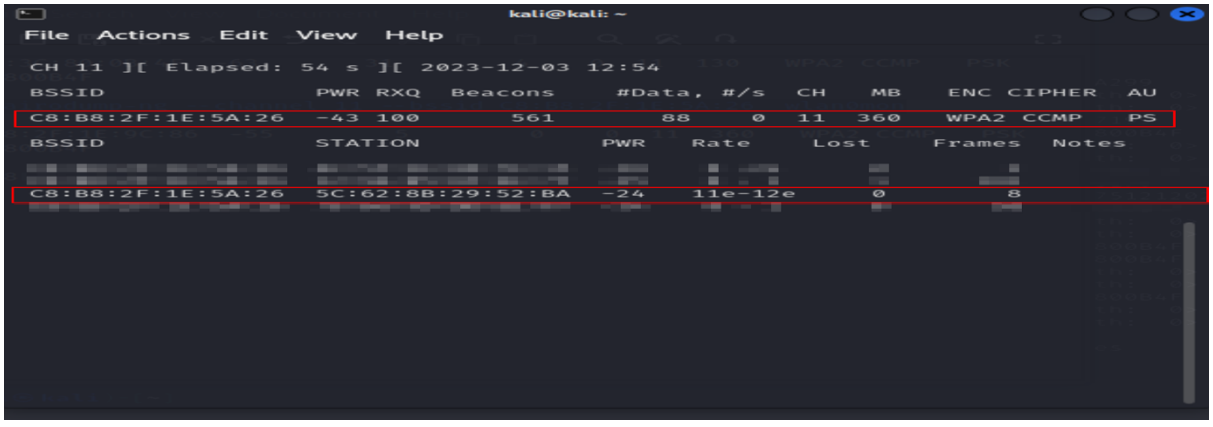

Figure 14: Target Mac address discovered.

Before Attack

Figure 15 shows the camera indicator light as green before the attack, meaning the camera is functioning and actively capturing the environment. The image shows in fig.. is also the result of camera activities before deauthentication attack.

Figure 15: Camera Indicator Light (Green)

Figure 16:Image from Camera before attack

III. Aireplay-ng

As part of the Aircrack-ng package, Aireplay-ng is an excellent tool for creating, injecting, and modifying wireless network traffic. It can aid with different stages of wireless security assessments or ethical hacking operations because it supports a variety of attack methods, such as deauthentication, false authentication, and ARP request injection.

In order to make the Tapo camera unusable with the wireless access point, we deauthenticate it using the "aireplay-ng" tool. The network camera may no longer be able to be viewed, which will endanger people, businesses, and network equipment at risk for security breaches. The procedure that "aireplay-ng" uses to deauthenticate the camera station from the wireless access point is depicted in the accompanying figure 17.

The command: sudo aireplay-ng --deauth 0 -a C8:B8:2F:1E:5A:26 -C 5C:62:8B:29:52:BA wlan0mon. The command literally tells aireplay to deauthenticate Tapo camera from the access point and make it unusable.

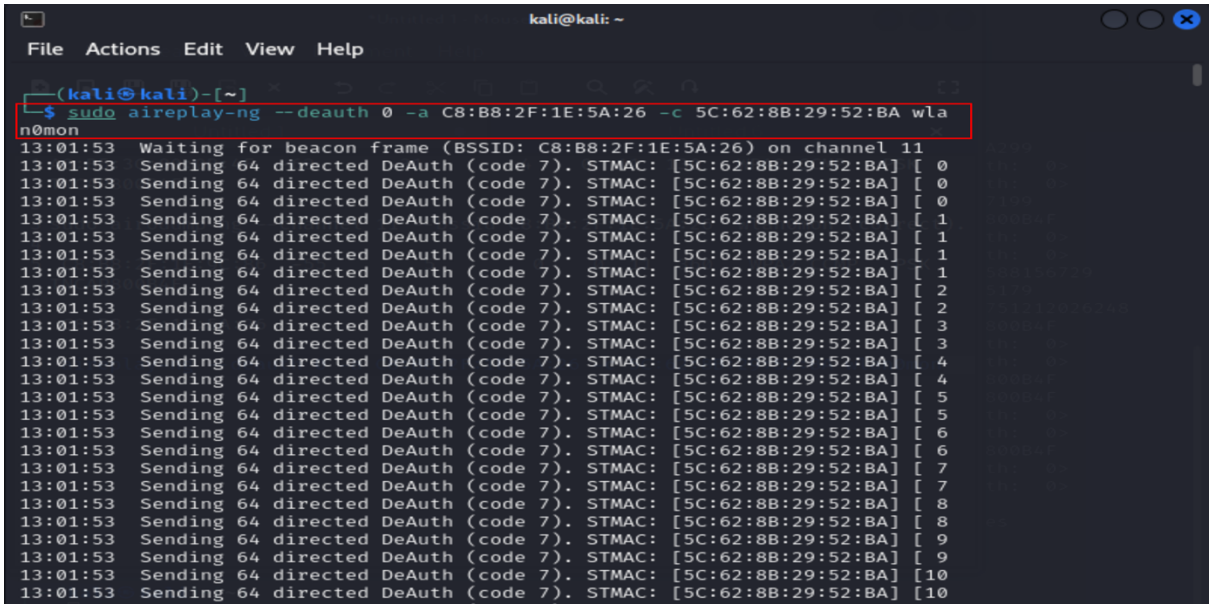

Figure 17: Deauthentication attack in progress

During Attack

The camera indication light turned red as the deauthentication procedure progressed forward. This modification indicates that viewers will no longer be able to see the camera's feed because it has been unplugged from the network. Figure 18 shows as the camera indicator light turn red and figure 19 shows user could not connect to the camera due to the attack.

Figure 18: Camera Light Indicator During attack (Red)

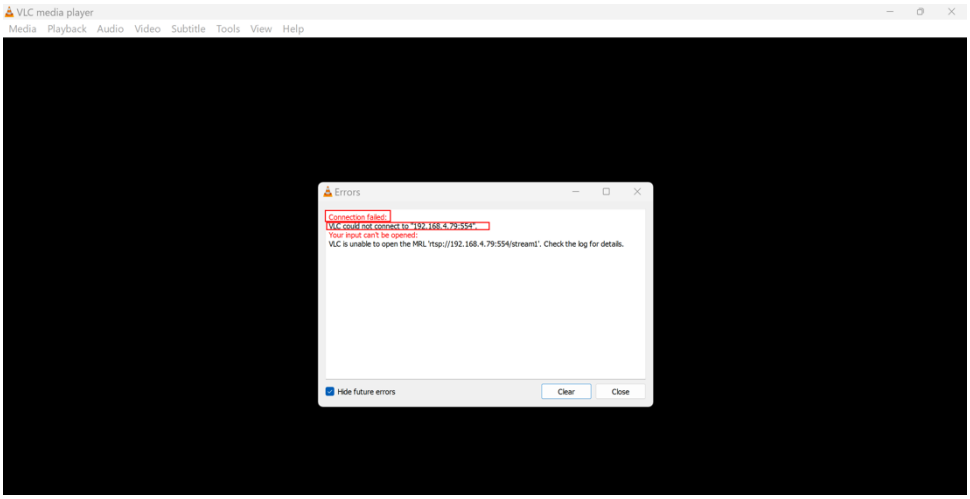

Figure 19: Service disrupt during attack.

4.4.1 Security Impact of Deauthentication Attack

The camera becomes inaccessible after successfully deauthenticating from the wireless access point. As a result of the attack, the camera's indicator light changes from green to red, indicating that the service has been disabled. Due to the camera's inability to establish connections, client access to it has been terminated. Deauthentication can have severe consequences, and detecting is challenging since there is no trace to pinpoint the source.

The camera's live stream and any related data will no longer be available to users who depend on it for monitoring or surveillance. Malicious actors may use a deauthentication attack to open a window of opportunity for illegal access or other security lapses while the camera is unplugged. The attack could also jeopardise the integrity of recorded data if the camera is part of a wider security ecosystem, resulting in gaps or inconsistent footage being preserved.

4.4.2 Mitigation Against Deauthentication Attack

The following are the mitigation measures needs to be implemented against deauthentication attack on TAPO Camera or any CCTV camera connected to wireless network.

1. Use of WPA3 Encryption: Use the most recent version of the Wi-Fi Protected Access (WPA3) encryption standard to ensure secure network communication between devices. When it comes to security features, WPA3 is superior to its predecessors.
2. Intrusion Detection System (IDS): Deploy intrusion detection systems capable of identifying patterns linked to deauthentication attempts. IDS can assist detect and address security issues by sending out notifications in real time.
3. Wireless Intrusion Prevention Systems (WIPS): To track and examine wireless network activity, implement WIPS. By implementing automated defenses, WIPS can recognize and lessen a variety of threats, including deauthentication assaults.

4. Physical Security: Ensure the physical security of camera is prevented from illegal tampering and accessing. The camera should be installed in a secured and restricted area.

5. Authentication Timeout: Set up your devices to automatically log off the network after a predetermined amount of time. As a result, attackers' window of opportunity to launch deauthentication assaults is reduced.

4.5 Raspberry Pi

Raspberry Pis are small single-board computers developed by the Raspberry Pi Foundation, a UK-based charity. To promote the teaching of basic computer science in schools and to provide a platform for hobbyists and developers to experiment with computing and electronics, these cheap, credit card-sized computers were developed. This device was used as part of the research as part of IOT hacking. A vulnerability in the device was exploited and successfully gained access to the device remotely. There is physical and software version of Raspberry Pi, in this research, a soft version was used to carried out the experiment. The version used for this exercise is Raspberry PI Imager V1.8.1 download from the official website of Raspberry using this link: https://www.raspberrypi.com/software/

4.5. 1 Penetration on the Raspberry PI Using SSH Credential

To exploit Raspberry Pi using SSH Credential can be carried using different methods, namely, nmap scripts, Metasploit's and hydra method to brute force the login credential. For this purpose, this penetration, Metasploit method was used to exploit the ssh login credential to get the username and password. To execute this attack, the following step would be followed.

Step 1: Scanning For Raspberry PI IP

To scan the for the raspberry network, nmap tool was used to check the network the device is connected. From the scanning, the result show list of available devices running on the network, device is running on virtualbox.

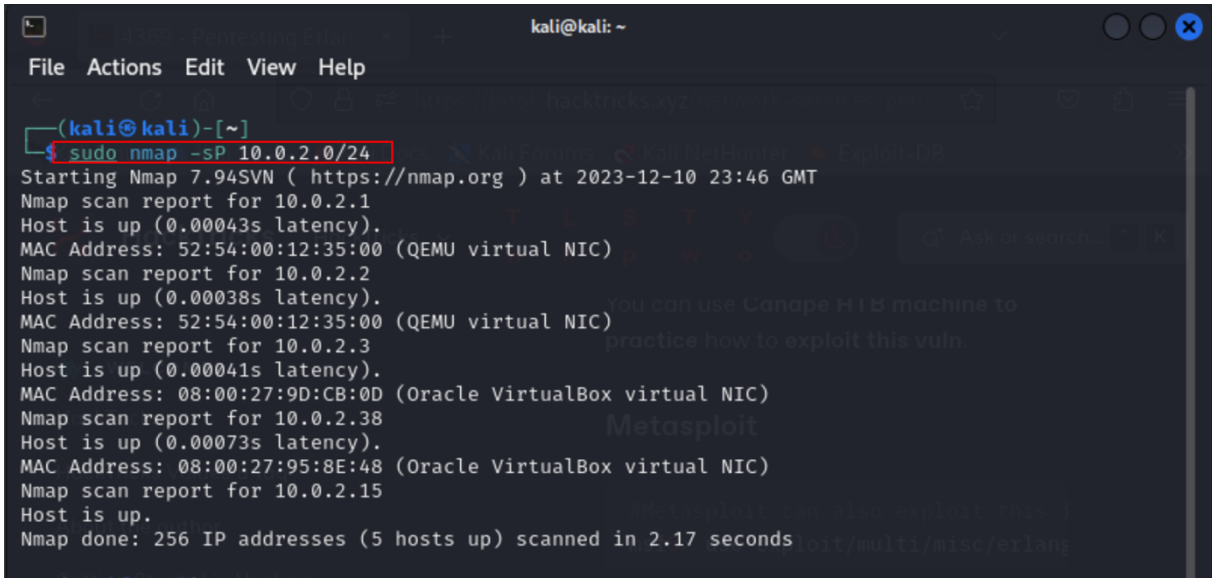

Figure 20: Scanning

To further find informaation about the target , nmap scan was use on all the device. Since the target is Raspberry Pi, the result is showned in the Figure…. Nmap scan reveal there is an open port on the device which is a tcp port 22 (SSH). This open could allow attacker to remotely connected to device.

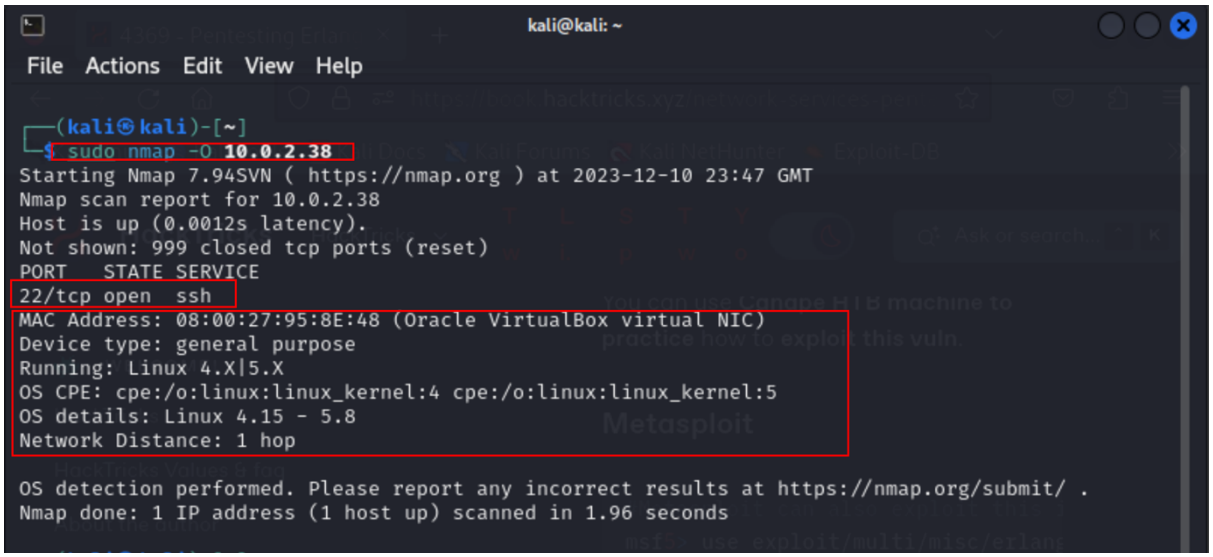

Figure 21: Raspberry Pi device information

Step 2: Using Metasploit

Users can find, exploit, and verify security flaws in computer systems with the use of Metasploit, an open-source penetration testing platform. It offers a thorough platform that security experts and ethical hackers may use to evaluate a network's security posture, spot any vulnerabilities, and gauge how well security solutions are working.

First before, implementing Metasploit, start the PostgreSQL database enabled the database for the attack by using this command: “sudo service postgresql start”. The next stage is to start the initiate Metasploit using “msfconsole” as shown in the fig…

Figure 22: Metasploit

To find the appropriate module, the search command was used. The command is “search ssh”. The result is highlighted in the figure 23. The ssh_login module is what is needed for the attack.

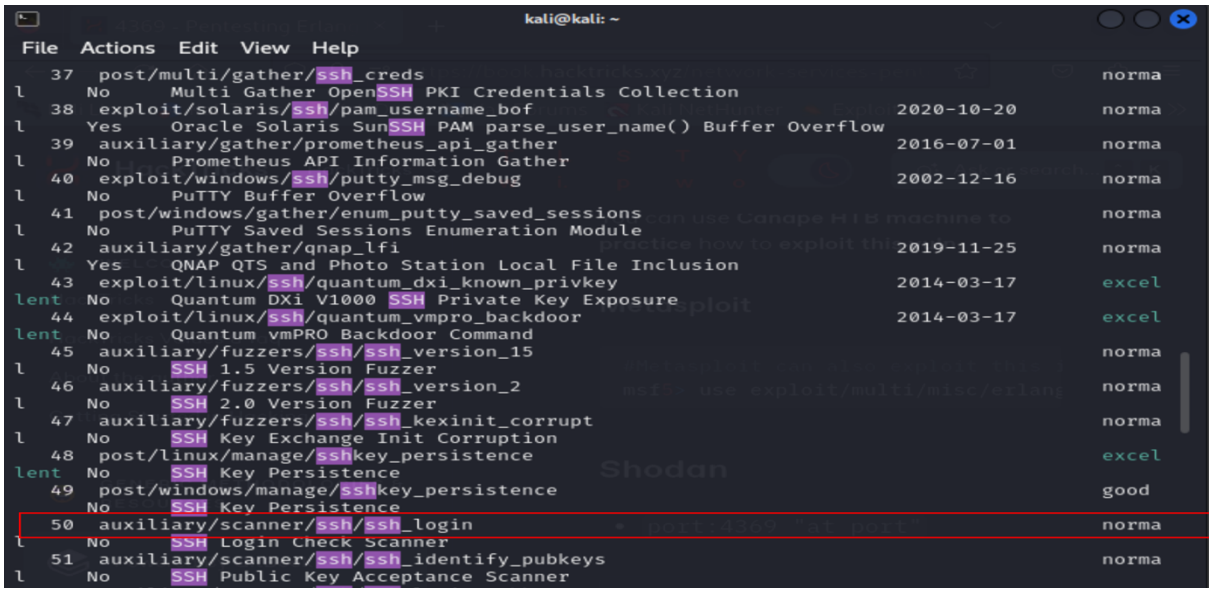

Figure 23: SSH module search

The next stage is use the “use command” options to display the available settings for the command as shown in the figure 24 below.

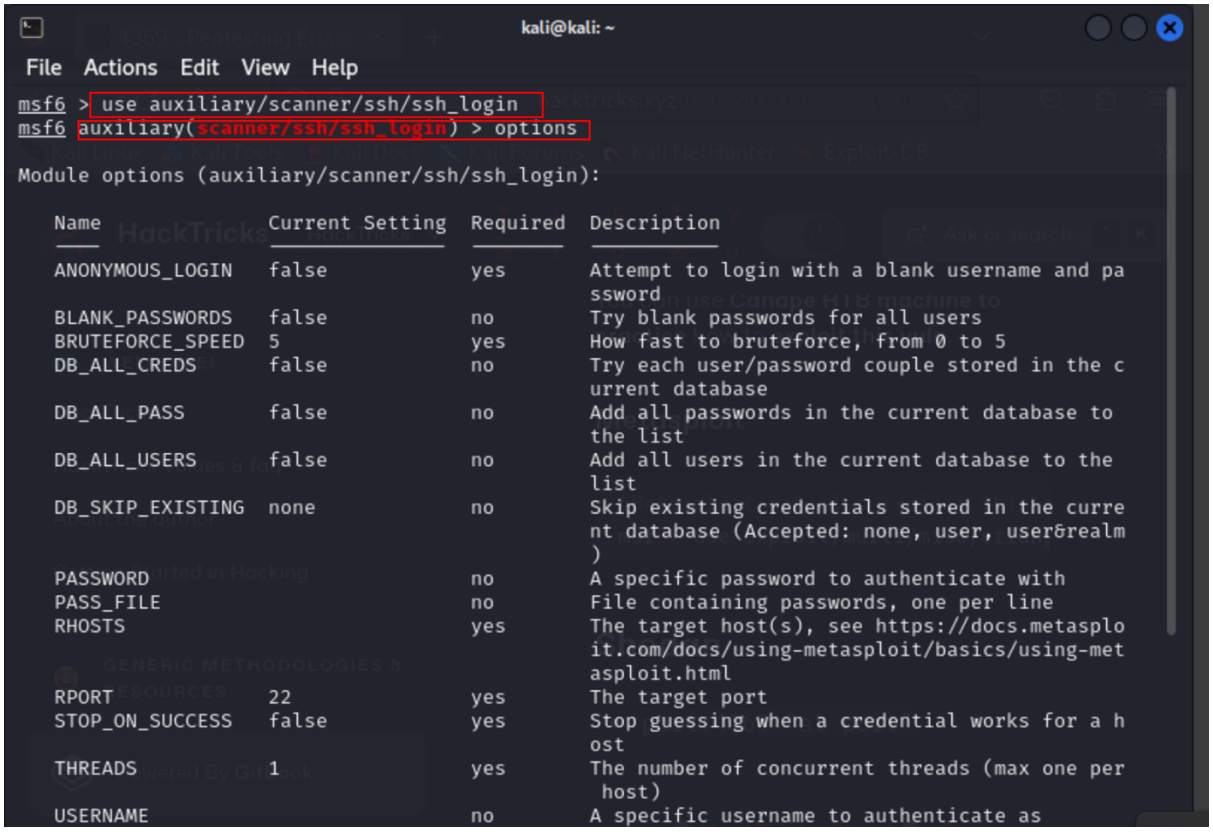

Figure 24: Setting the scanner

For this attack to work, a RHOSTs, user_file, pass_file, versbose and others would be set to carried out the attack

1. “set rhosts 10.0.2.38” is the first IP address of the target
2. “set stop_on_success true” is to stop when a valid username and password is found.
3. “set user_file username.txt” settting the username list.
4. “set pass_file password.txt” setting the password list.
5. “set verbose true” display all attempt.
6. “run” to start the attack.

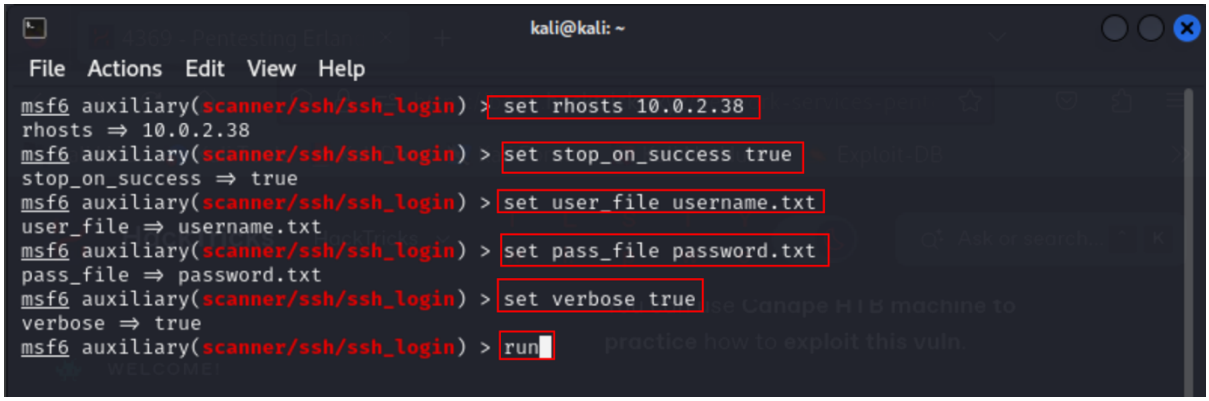

Figure 25:Payload configuration

The figure 26 shows the attack was successfully with discovery of the username “admin” and password “admin”. The image also display a session has been created.

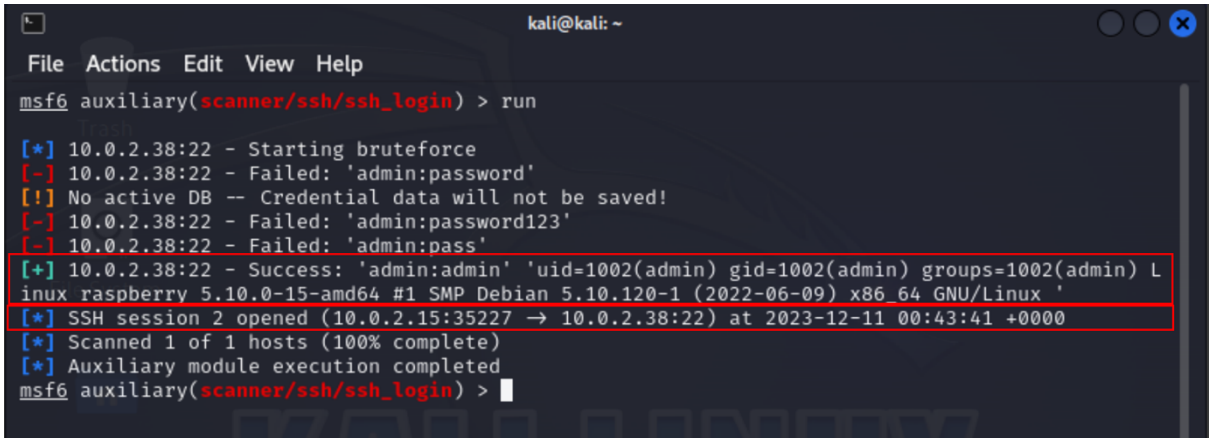

Figure 26: Credential hacked

Step 3: Gaining Access

After successfully hacked the ssh-login credential and a session is created, the attacker further checked the user with opening session.

Using “sessions command“ would display the active session as shown in the figure below. To interact with shell, the -i flag was alongside using this command: “sessions -i 3”.

After opened the session, the attacker have access to the device and act as the legimimate user after gaining access. The following command were used be the attacker.

1. Whoami display who the active user is which “admin”
2. Pwd dispplay current working directory and return “/home/admin”
3. Passwd which would enable changing of password as shown in the figure. The command requesting for current password and also new password.

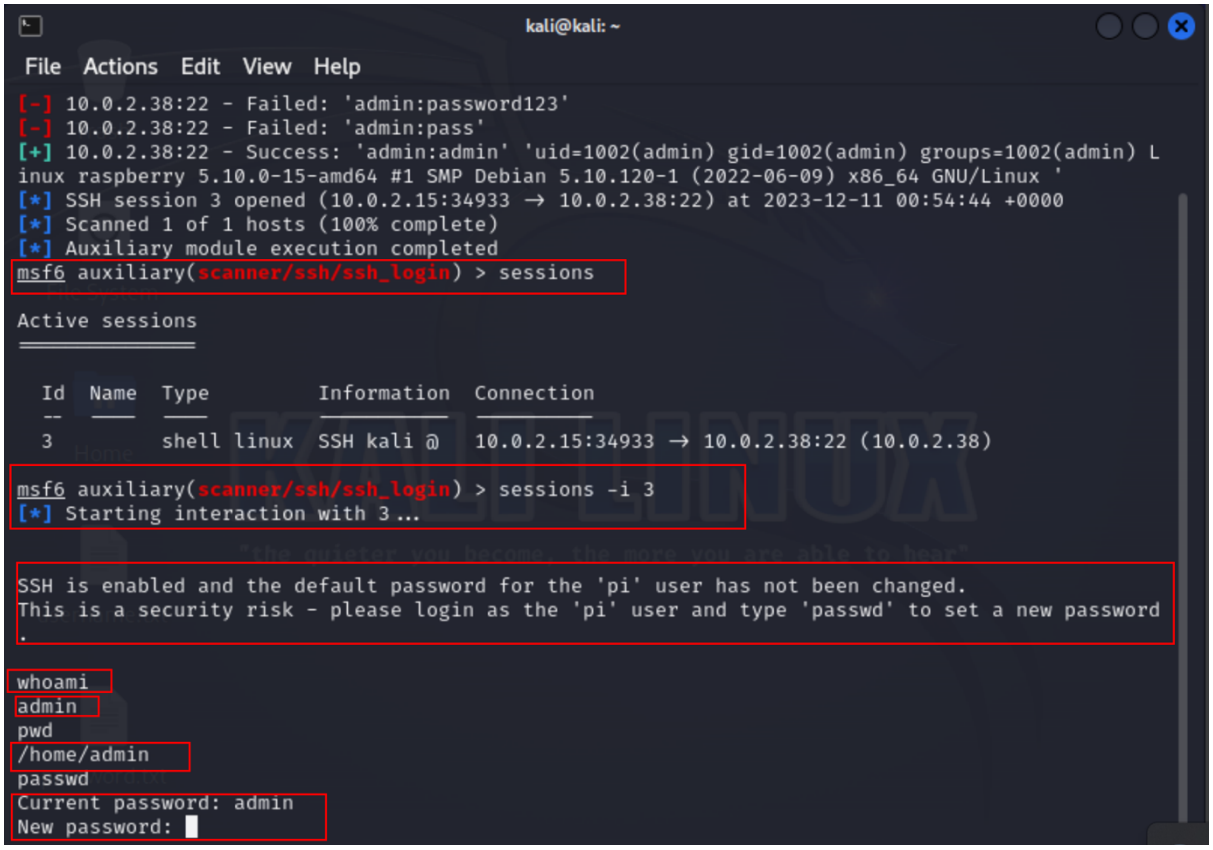

Figure 27: Gaining access to Raspberry Pi

Testing SSH Connection From Another Device

The attacker test the ssh connection using another method after hacked the username and password without using the shell interaction. The attacker was able to connect to the device remotely. The figure 28 shows the attacker have access to “admin@raspberry” device.

Figure 28: SSH connection successful

4.5. 3 Impact Of The Attack On Raspberry PI

Hacking a Raspberry pi through SSH could pose a significant consequence on the device. The following are the potential effects of attack on Raspberry pi.

1. Data theft and Manipulation: An attacker may try to alter or steal private information that is already stored on the Raspberry Pi. Personal data, login credentials, and any other information pertinent to the use of the device or related projects may fall under this category.
2. Resource Abuse: Raspberry Pi devices that have been compromised can be used for resource abuse, such as cryptocurrency mining scripts and botnets. The device's performance may be degraded as a result of increased resource use and higher electricity bills.
3. Network Exploitation: An attacker could attempt to investigate and take advantage of other devices on the same network once they are inside the Raspberry Pi. This could entail trying to migrate laterally, looking for security holes, and even breaking into other systems.

4.5.4 Mitigation Techniques Against Raspberry Pi

The following techniques could prevent SSH login credentials hacking, attacker could successfully hacked the credentails but this extra layer of security could prevent the attacker to login into the device.

1. Two Factor Authentication: Enable two factor authentication (2FA) on raspberry PI would provides extra security measures that will prevent unauthorized access to the device. Two-factor authentication (2FA) adds another layer of protection by ensuring the user has the right password and access to a physical device or authentication app. We'll take the next action to make two-factor authentication available. To implement two factor authentication on the Raspberry PI, Google authenticator would be installed on the device as shown in the figure 29.

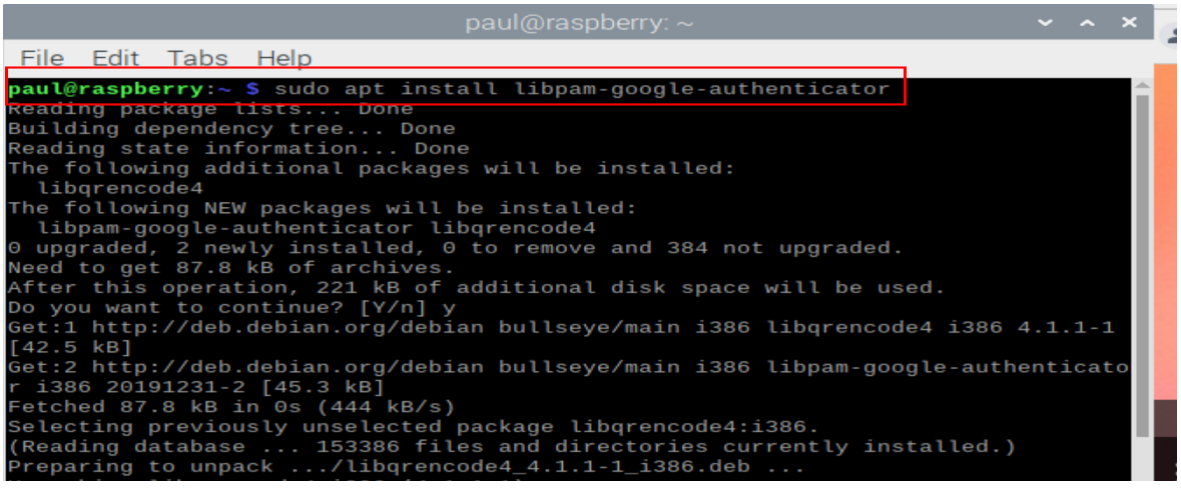

Figure 29:Google authentication installation

After Google Authenticator was installed, an Android smartphone was connected to a QR code that was created. To create a connection between the device and the mobile device for token authentication, the generated QR code, as shown in the figure, was utilised.

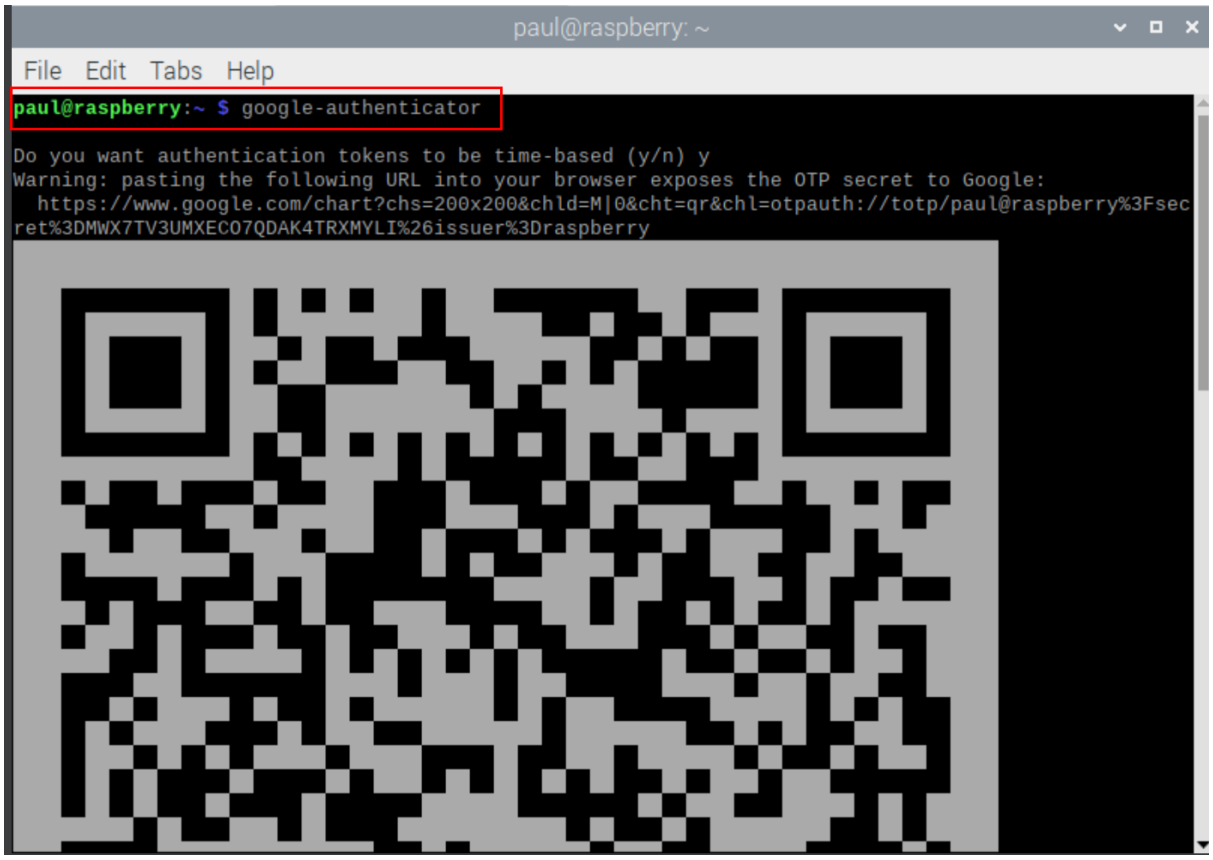

Figure 30: QR Code Connection

Mobile token for verification. This token is required for user to have access to the device. The image shown the time-based google authenticator code.

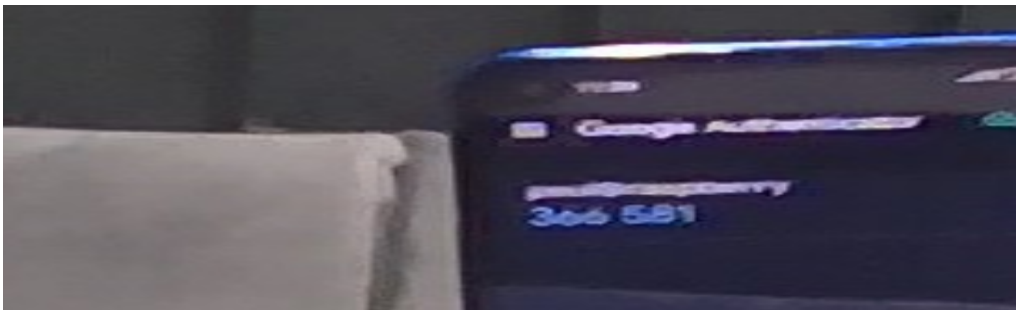

Figure 31: Verification code

This figure 32 shows an attacker trying to connect via SSH to a Raspberry Pi. As a result of additional security measures, the attacker cannot proceed beyond the initial access stage. As a result, the attacker does not know about these measures or lacks the credentials needed to verify them. The attacker passes the first authentication stage with the correct

username and password, but fails to progress to the second authentication stage, which involves a token verification.

Figure 32: Verification required by SSH login

2. Disable SSH Root Login: Disable A system's security can be enhanced by disabling SSH root login because it narrows the attack surface, increases accountability, lessens the impact of brute force attacks, and upholds the least privilege principle.

Figure 33: Disable SSH Root login

3. Implementing Firewalls: Installing firewalls on the device would enhanced the security of the Raspberry device. The firewalls would block suspicious traffic coming from unknown sources. Firewall can be configure by using physical and software configuration. A software configurarion can be done using a set of rules as shown in figure …..

Rule:sudo iptables -A INPUT -p tcp -s 10.0.2.0/24 --dport 22 -m conntrack --ctstate NEW,ESTABLISHED -j DROP

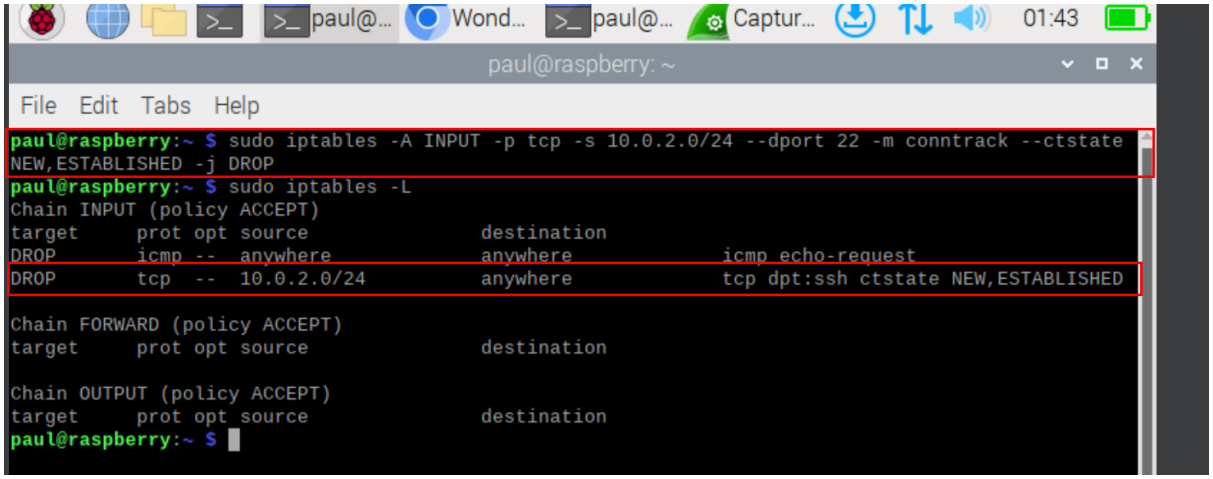

Figure 34: Firewall

The illustration shows an attacker attempting to establish a remote SSH connection in order to obtain access to the user "paul@raspberry". Nevertheless, the device's active firewall blocks the intrusion. The snapshot also shows that the attacker made multiple unsuccessful attempts before the connection timed out.

Figure 35: Connection blocked by firewall

The Raspberry Pi's firewall rules are implemented as seen in the figure, which denies an SSH connection from the IP address range 10.0.2.0/24 since it appears to be related to questionable login behaviour. The packet is dropped by the firewall since it does not fulfil the defined rule's criteria, as shown by the Wireshark capture.

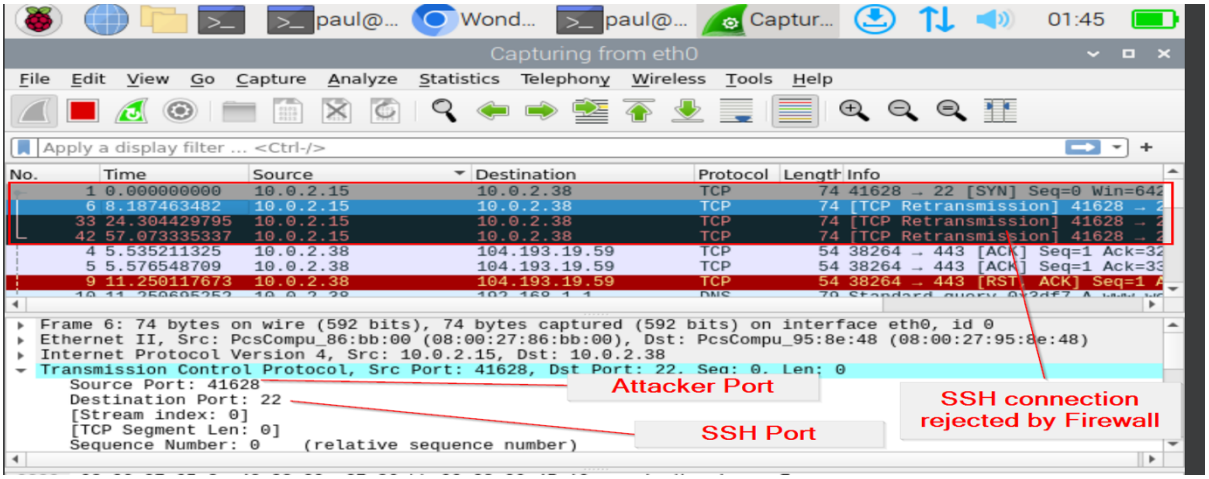

Figure 36: Wireshark Analysis of reject packets by firewall

4.6 Discussion

The comfort and efficiency that Internet of Things (IoT) gadgets provide to our daily lives have made them indispensible. But the widespread use of these gadgets brings up important issues that need to be carefully considered. The most important of these worries is the absolute necessity of strong security measures. It would be dangerous to overlook the security elements of IoT devices, especially as technology develops and hackers use more advanced techniques to compromise these devices. Considering the possibility of malevolent actors exploiting flaws, the security of Internet of Things devices is critical. Technology advances, and with it do the techniques used by cybercriminals to compromise these devices. Attackers are unrelenting in their use of a variety of tools to wreck havoc on governments, businesses, individuals, and society at large.

TAPO camera and Raspberry Pi vulnerabilities were found during a penetration exercise, indicating possible points of exploitation for attackers. In the absence of sufficient security measures, these weaknesses might be exploited with dire repercussions. The study effectively illustrated how hackers may take control of IoT devices and use them for their own benefit, resulting in a variety of negative effects. Numerous vulnerabilities in the devices were found during the penetration testing, most of which users choose to ignore. These disregarded vulnerabilities offer possible ports of entry that hackers could use to their advantage. The study emphasises how crucial it is to fix these issues in order to protect the devices and their users.

Considering the enormity of the issues that were found, the researcher provides mitigating strategies to avoid reoccurring occurrences. To strengthen the security of IoT devices and reduce the risks posed by potential cyber threats. Users, manufacturers, and legislators must all engage pro-actively. By doing this, we can make sure that our security and privacy are maintained while the advantages of IoT technology continue to improve our lives.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5.1 Conclusion

This report extensively examines the security vulnerabilities present in IoT devices and emphasizes the urgent need for heightened security measures. The analysis highlights the potential threats associated with these vulnerabilities and demonstrates them through successful penetration testing on the TP-Link Tapo Wi-Fi camera and Raspberry Pi. As IoT devices become indispensable in our daily lives, the report stresses the critical importance of addressing security concerns, especially with the evolving sophistication of cyber threats.

The study reveals vulnerabilities in the Tapo security camera and Raspberry Pi by indicating possible points of exploitation for malicious actors, and without adequate security measures, these weaknesses could lead to severe consequences. The research effectively illustrates how hackers could exploit these devices, emphasizing the significance of addressing disregarded vulnerabilities that users often overlook.

Given the gravity of the identified issues, the report provides mitigating strategies to prevent recurring occurrences. It calls for a collaborative effort involving users, manufacturers, and legislators to proactively engage in strengthening IoT device security and mitigating the risks posed by potential cyber threats. The overarching goal is to ensure the continued improvement of IoT technology while safeguarding security and privacy for all users.

5.2 Future work

The scope of the thesis was limited due to time constraints. However, to improve the project, potential future work includes analysing more IoT devices used daily such as smartwatches, smart TVs, smartphones, and so on. Additionally, untested attacks such as MiTM attacks on both devices, DNS cache poisoning, and IP spoofing can be performed on the Raspberry Pi device.

REFERENCES

1. Abdymanapov, S., Barlybaev, A.A. and Б.А. Алтынбек (2022) InfoSec Risk Assessment Methodology based on the example of Learning Management Systems Analysis. *Қарағанды университетінің хабаршысы* 107 (3), 84–95.
2. Alamareen, A., Malak, H. and Abuasal, S. (2023) *Cyber Security & IoT Vulnerabilities Threats Intruders and Attacks Research Review*. *Journal of Namibian Studies* 2197–5523.
3. Arreaga, N., Enriquez, G.M., Clavero, S. and Estrada, R. (2023) Security Vulnerability Analysis for IoT Devices Raspberry Pi using PENTEST. *Procedia Computer Science* 224, Elsevier BV223–230.
4. Awang, N.F., Zainudin, A.F.I.M., Marzuki, S., Alsagoff, S.N., Tajuddin, T. and Jarno, A.D. (2021) Security and Threats in the Internet of Things Based Smart Home. *Lecture Notes on Data Engineering and Communications Technologies* 676–684.
5. Bosch Global (n.d.) *Automated Valet Parking – fast, safe, driverless*. https://www.bosch.com/stories/automated-valet-parking/#:~:text=Automated%20valet%20parking%20allows%20the Accessed 4 August 2023.
6. Chandan, A.R. and Khairnar, V.D. (2018) Bluetooth Low Energy (BLE) Crackdown Using IoT.
7. Check Point Research (2023) *The Tipping Point: Exploring the Surge in IoT Cyberattacks Globally*. https://blog.checkpoint.com/security/the-tipping-point-exploring-the-surge-in-iot-cyberattacks-plaguing-the-education-sector/#:~:text=Highlights%3A Accessed 6 July 2023.
8. Cisco (2020) *Cisco Annual Internet Report (2018–2023) White Paper*. https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html Accessed 10 July 2023.
9. Clincy, V. and Shahriar, H. (2019) *IoT Malware Analysis*. 920–921https://ieeexplore.ieee.org/abstract/document/8754110 Accessed.

10. Contos, B. (2023) *Council Post: The Secret, Insecure Life Of Security Cameras*. https://www.forbes.com/sites/forbestechcouncil/2023/03/01/the-secret-insecure-life-of-security-cameras/ Accessed 8 June 2023.
11. DDS Foundation (n.d.) *What is DDS?* https://www.dds-foundation.org/what-is-dds-3/ Accessed 2 June 2023.
12. Duarte, F. (2023a) *Amount of data created daily (2023)*. https://explodingtopics.com/blog/data-generated-per-day Accessed 7 July 2023.
13. Duarte, F. (2023b) *Number of IoT Devices (2023-2030)*. https://explodingtopics.com/blog/number-of-iot-devices# Accessed 16 August 2023.
14. EC Council (2024) *What is Cyber Threat Modeling | Importance of Threat Modeling*. https://www.eccouncil.org/threat-modeling/#:~:text=VAST%20(Visual%2C%20Agile%2C%20and Accessed 23 January 2024.
15. EC-Council (2022a) *DREAD Threat Modeling: An Introduction to Qualitative Risk Analysis*. https://www.eccouncil.org/cybersecurity-exchange/threat-intelligence/dread-threat-modeling-intro/ Accessed 8 September 2023.
16. EC-Council (2022b) *DREAD Threat Modeling: an Introduction to Qualitative Risk Analysis*. https://www.eccouncil.org/cybersecurity-exchange/threat-intelligence/dread-threat-modeling-intro/ Accessed.
17. EC-Council (2022c) *What Is Penetration Testing? Strategic Approaches and Types*. https://www.eccouncil.org/cybersecurity-exchange/penetration-testing/penetration-testing-strategic-approaches-types/ Accessed.
18. Engebretson, P. (2014) *Introdução ao Hacking e aos Testes de Invasão*. Novatec Editora.
19. Fortinet (2023) *What is Brute Force Attack? | Definition, Types & How It Works*. https://www.fortinet.com/resources/cyberglossary/brute-force-attack Accessed 10 July 2023.
20. Gerodimos, A., Maglaras, L., Ferrag, M.A., Ayres, N. and Kantzavelou, I. (2023) IoT: Communication protocols and security threats. *Internet of Things and Cyber-Physical Systems* .
21. Gustafsson, H. and Kvist, H. (2022) *Cyber Security Demonstrations using Penetration Testing on Wi-Fi Cameras*.

22. HaddadPajouh, H., Dehghantanha, A., M. Parizi, R., Aledhari, M. and Karimipour, H. (2021) A survey on internet of things security: Requirements, challenges, and solutions. *Internet of Things* 14, 100129.
23. Hassan, M. (2023) *'Demonstration of cyber security through Penetration testing on IP camera' DETECTING PHISING WEBSITES USING MACHINE LEARNING View project*.
24. Hojlo, J. (2021) *Future of Industry Ecosystems: Shared Data and Insights*. https://blogs.idc.com/2021/01/06/future-of-industry-ecosystems-shared-data-and-insights/ Accessed 24 June 2023.
25. HP (2018) *Internet of Things Security Primer*. HPhttps://www.hp.com/us-en/shop/tech-takes/internet-of-things-security-primer Accessed 4 September 2023.
26. Javed , M.H. (2023) *Internet of Things Hacking: Ethical Hacking of a Smart Camera Dissertation Project Report*.
27. Jester, T. (2023) *Understanding RockYou.txt: A Tool for Security and a Weapon for Hackers*. https://www.keepersecurity.com/blog/2023/08/04/understanding-rockyou-txt-a-tool-for-security-and-a-weapon-for-hackers/#:~:text=txt- Accessed 9 July 2023.
28. Kandasamy, K., Srinivas, S., Achuthan, K. and Rangan, V.P. (2020) IoT cyber risk: a holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process. *EURASIP Journal on Information Security* 2020 (1),.
29. Kaspersky (2021) *Ransomware Attacks and Types – How Encryption Trojans Differ*. https://www.kaspersky.com/resource-center/threats/ransomware-attacks-and-types Accessed 1 September 2023.
30. MacBride, E. (2023) *The dark web's criminal minds see Internet of Things as next big hacking prize*. https://www.cnbc.com/2023/01/09/the-dark-webs-criminal-minds-see-iot-as-the-next-big-hacking-prize.html# Accessed 14 August 2023.
31. Mahlous, A.R. (2023) Threat model and risk management for a smart home IoT system. *Informatica* 47 (1),.
32. Mallik, A. (2019) MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE WORDS. *Cyberspace: Jurnal Pendidikan Teknologi Informasi* 2 (2), 109.
33. McAfee (n.d.) *What is malware and why do cybercriminals use malware?* https://www.mcafee.com/en-gb/antivirus/malware.html Accessed 9 June 2023.

34. Microsoft (2010) *Improving Web Application Security: Threats and Countermeasures*. https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10) Accessed 19 February 2023.
35. Microsoft (2023) *Top 5 Most Famous DDoS Attacks*. https://www.microsoft.com/en-us/microsoft-365-life-hacks/privacy-and-safety/top-5-most-famous-ddos-attacks Accessed 8 June 2023.
36. Mira, F. and Izzat Alsmadi (2019) Review of Analysis on IoT Components, Devices and Layers Security.
37. Mirror Review (2021) *Rise in Cyber-attack: firmware a big issue for company*. https://www.mirrorreview.com/rise-in-cyber-attack-firmware/ Accessed 8 June 2023.
38. MKTG1, S. (2023) *The OWASP IoT top 10 vulnerabilities and how to mitigate them | SISA Blog*. https://www.sisainfosec.com/blogs/the-owasp-iot-top-10-vulnerabilities-and-how-to-mitigate-them/ Accessed 21 June 2023.
39. None Asma'a Alamareen, Malak, N. and None Sara Abuasal (2023) Cyber Security & IoT Vulnerabilities Threats Intruders and Attacks Research Review. *Journal of Namibian Studies : History Politics Culture* 33,.
40. O'Neill, M. (2016) Insecurity by Design: Today's IoT Device Security Problem. *Engineering* 2 (1), 48–49.
41. Oracevic, A., Dilek, S. and Ozdemir, S. (2017) Security in internet of things: A survey. *2017 International Symposium on Networks, Computers and Communications (ISNCC)* .
42. Petrosyan, A. (2023) *Annual number of IoT attacks global 2022*. https://www.statista.com/statistics/1377569/worldwide-annual-internet-of-things-attacks/ Accessed 2 August 2023.
43. Qureshi, A. 2023. Lecture 2: Introduction to Ethical Hacking notes. Lecture notes. Ethical Hacking COS7029-B. University of Bradford. Delivered 06 February 2023.
44. Qureshi, A. 2023. Lecture 3: Network Scanning. Lecture notes. Ethical Hacking COS7029-B. University of Bradford. Delivered 12 February 2023.
45. Radholm, F. and Abefelt, N. (2020) *Ethical Hacking of an IoT-device: Threat Assessment and Penetration Testing*.

46. Roberts, M.C. (2020) *Internet of things Security Penetration Testing*. Cardiff University:
47. Russell, B. and Van Duren, D. (2016) *Practical Internet of things security : a practical, indispensable security guide that will navigate you through the complex realm of securely building and deploying systems in our IoT-connected world*. Birmingham: Packt Pub.
48. Security, M. (2007) *STRIDE chart*. https://www.microsoft.com/en-us/security/blog/2007/09/11/stride-chart/ Accessed 8 September 2023.
49. Shanley, A. and Johnstone, M. (2015) Selection of penetration testing methodologies: A comparison and evaluation. *Selection of penetration testing methodologies: A comparison and evaluation* 65–72.
50. Shokoufeh Seifi, Beaubrun, R., Bellaiche, M. and Halabi, T. (2023) A Study on the Efficiency of Intrusion Detection Systems in IoT Networks.
51. Sivapriyan, R., Sushmitha, S.V., Pooja, K. and Sakshi, N. (2021) *Analysis of Security Challenges and Issues in IoT Enabled Smart Homes*. 1–6https://ieeexplore.ieee.org/document/9683324 Accessed.
52. Srivastava, A., Gupta, S., Quamara, M., Chaudhary, P. and Aski, V.J. (2020) Future IoT-enabled threats and vulnerabilities: State of the art, challenges, and future prospects. *International Journal of Communication Systems* 33 (12), e4443.
53. Stallings, W. and Lawrie Brown (2017) *Computer security*. S.L.: Pearson Education (Us.
54. Tidy, J. (2022) Ukrainian power grid 'lucky' to withstand Russian cyber-attack. *BBC News* 12 April.
55. Vailshery, L. (2022) *IoT connected devices worldwide 2019-2030*. https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/ Accessed 4 August 2023.
56. Wang, C. (2018) *The 2016 Dyn Attack and its Lessons for IoT Security | MS&E 238 Blog*. https://mse238blog.stanford.edu/2018/07/clairemw/the-2016-dyn-attack-and-its-lessons-for-iot-security/ Accessed 17 July 2023.
57. Whitman, M.E. and Mattord, H.J. (2018) *Principles of information security*. 7th Edition. Boston, Mass.: Cengage Learning.

58. Williams, P., Dutta, I.K., Daoud, H. and Bayoumi, M. (2022) A Survey on Security in Internet of Things with a Focus on the Impact of Emerging Technologies. *Internet of Things* 19, 100564.
59. Xu, H., Ding, J., Li, P., Zhu, F. and Wang, R. (2018) A Lightweight RFID Mutual Authentication Protocol Based on Physical Unclonable Function. *Sensors* 18 (3), 760.
60. Yavwa, Y. (2000a) *The Firewall Technology* .
61. Yavwa, Y. (2000b) *The Firewall Technology 1 Executive Summary*.

APPENDIX I

Dread Threat Methodology rating Matrix

Damage Potential: How Much Damage Could the Attack Cause?

- 0: No damage
- 5: Information disclosure
- 8: Non-sensitive user data related to individuals or employer compromised
- 9: Non-sensitive administrative data compromised
- 10: Destruction of an information system; data or application unavailability

Reproducibility: How Easily Can the Attack Be Reproduced?

- 0: Difficult or impossible
- 5: Complex
- 7.5: Easy
- 10: Very easy

Exploitability: What's Required to Launch the Attack?

- 2.5: Advanced programming and networking skills
- 5: Available attack tools
- 9: Web application proxies
- 10: Web browser

Affected Users: How Many People Would the Attack Affect?

- 0: No users
- 2.5: Individual user
- 6: Few users
- 8: Administrative users
- 10: All users

Discoverability: How Easy Is the Vulnerability to Discover?

- 0: Hard to discover the vulnerability
- 5: HTTP requests can uncover the vulnerability
- 8: Vulnerability found in the public domain
- 10: Vulnerability found in web address bar or form

Overall Threat Rating

The overall threat rating is calculated by summing the scores obtained across these five key areas. The risk severity categories for a threat are as follows:

- **Critical (40–50):** Critical vulnerability; address immediately.
- **High (25–39):** Severe vulnerability; consider for review and resolution soon.
- **Medium (11–24):** Moderate risk; review after addressing severe and critical risks.
- **Low (1–10):** Low risk to infrastructure and data.