OULUN AMMATTIKORKEAKOULU

ENHANCING IOT SECURITY IN SMART HOMES USING KALI LINUX FOR PENETRATION TESTING

Esther Fatoyinbo & Sujata Shrestha
Bachelor's Thesis
Spring 2025
Degree Programme in Information Technology
Oulu University of Applied Sciences

ABSTRACT

Oulu University of Applied Sciences
Degree Programme in Information Technology
Option of Software Engineering

Authors: Esther Fatoyinbo & Sujata Shrestha
Title of thesis: Enhancing IoT Security in Smart Homes Using Kali Linux for Penetration Testing
Thesis supervisor: Ville Majava
Term and year of completion: Spring 2025
Pages: 55

This thesis examines the security challenges related to Internet of Things (IoT) devices utilized in smart homes. The swift expansion of these beneficial devices has markedly improved convenience, automation, and energy efficiency. The world is expected to see an increased prevalence of IoT systems in the forthcoming years. This increasing interconnectivity is accompanied by considerable cybersecurity challenges. This research aimed to examine the security status of select consumer-grade IoT devices via ethical penetration testing utilizing Kali Linux. The study concentrated on the Airam SmartHome temperature and humidity sensor and a generic humidity sensor, bought online, simulating realistic attack scenarios within a controlled setting.

This study employed tools including Nmap, Hydra, Metasploit, Wireshark, and Tcpdump to identify critical vulnerabilities, such as open and misconfigured services, outdated firmware, weak authentication, and unsecured communication protocols. Through the application of Service Enumeration, Operating System (OS) fingerprinting, brute-force testing, and packet capture techniques, the study identified methods by which malicious actors could exploit these systems.

Significant findings indicate the common occurrence of outdated Linux kernels, unencrypted web interfaces, and the use of legacy protocols such as Universal Plug and Play (UPnP) and Point-to-Point Tunneling Protocol (PPTP). Manufacturers, users, researchers, and policymakers are advised to enhance IoT security through secure-by-design engineering, firmware patching, and the implementation of robust authentication measures. This study highlights the necessity of ongoing security evaluations and user awareness to protect the integrity and privacy of smart home systems.

CONTENTS

1 INTRODUCTION

The swift advancement of the Internet of Things (IoT) industry has transformed the manner in which individuals engage with technology in their everyday activities. This advancement facilitates the integration of residences with various smart devices and appliances, enabling centralized control and monitoring via smartphones or tablets. Intelligent thermostats, lighting systems, smart refrigerators, and entertainment units have become increasingly commonplace, attributed to the flexibility, convenience, and efficiency they offer. (Das et al., 2023).

The effectiveness of these systems stem from their capacity to facilitate remote control and monitoring through a network of interconnected devices that utilize wireless protocols, including Wi-Fi, Zigbee, and Bluetooth for communication. The integration of seamless connectivity has markedly altered daily life, streamlined home management, and enhanced overall efficiency in lifestyles. While IoT facilitates improved automation and operational efficiency, it concurrently presents numerous security vulnerabilities and privacy risks. Unauthorized data access, device spoofing, and Man-in-the-Middle (MitM) attacks represent significant cyber threats to IoT systems (Ahmed & Khan, 2023). The increasing prevalence of IoT devices presents significant risks to individuals and businesses, potentially resulting in data breaches and system compromises.

This thesis examines security issues and emphasizes the significance of penetration testing with Kali Linux in detecting vulnerabilities in the security architecture of IoT devices, thus enhancing security in smart home environments.

Overview of IoT Smart Homes

The IoT revolution has significantly altered modern day households by combining various devices, including thermostats, security cameras, smart locks, lighting systems, and voice assistants, into cohesive ecosystems. These devices facilitate automation and remote control via wireless connectivity, thereby improving convenience, security, and energy efficiency (Atzori et al., 2010).

According to Adnyana et al. (2023), this increasing interdependence has revealed substantial cybersecurity challenges for smart homes because these devices capture, transmit, and process sensitive data, rendering them attractive targets for malicious actors.

Security cameras and environmental sensors, including temperature and humidity monitors, are among the most used smart home devices. Security cameras improve surveillance; however, numerous devices remain at risk due to default login credentials, unencrypted video feeds, and outdated firmware (Javed, 2023). Sensors for HVAC frequently exhibit poor encryption and insufficient authentication, resulting in exploitable vulnerabilities.

According to Reddy et al. (2022), common issues in smart home security involve the utilization of default usernames and passwords, which are susceptible to brute-force attacks. Moreover, numerous devices depend on unencrypted protocols like MQTT and CoAP, which enables attackers to intercept and manipulate traffic. Restricted processing power limits the application of sophisticated security features, thereby leaving devices vulnerable (Shafiq et al., 2022).

Furthermore, many manufacturers do not provide prompt firmware updates to address identified vulnerabilities. This delay heightens the risk of malware installation, device hijacking, or botnet recruitment for Distributed Denial-of-Service (DDoS) attacks (De Neira et al., 2023). The discovered concerns emphasize the necessity for enhanced security protocols in smart home settings.

Importance of Cybersecurity in IoT Environments

IoT system vulnerabilities can result in privacy breaches, financial losses, and potentially life-threatening scenarios. Ensuring cybersecurity in IoT environments is crucial for protecting personal data, maintaining device integrity, and preventing unauthorized access by malicious actors.

The Mirai botnet attack demonstrates these risks, as attackers used weak IoT credentials to seize thousands of devices and perform a significant DDoS attack, disrupting major internet services (Antonakakis et al., 2017). The incident brought

out the necessity for improved authentication, secure protocols, and prompt updates.

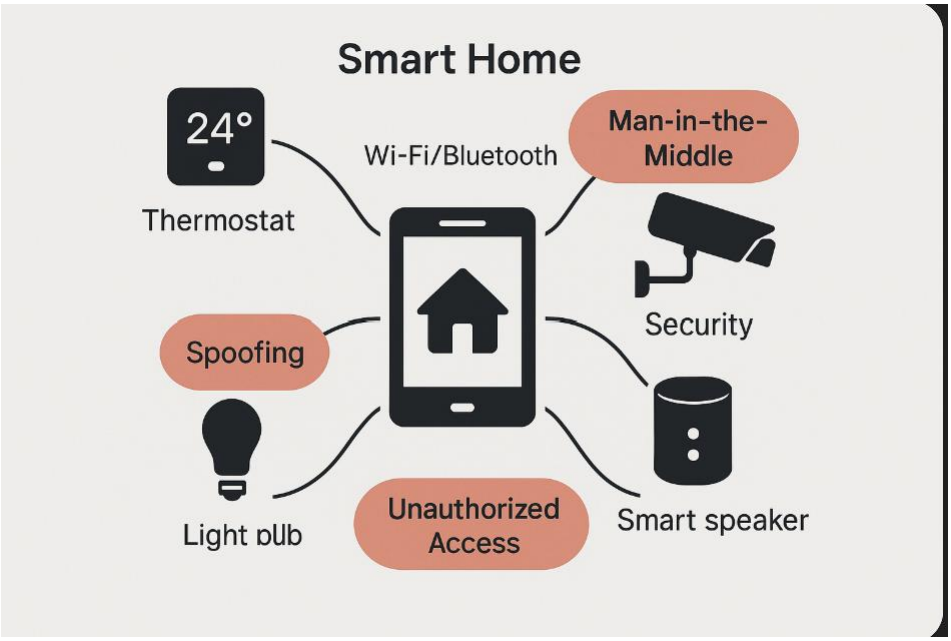


Figure 1.IoT Smart Home Devices and Security Challenges. An image created with AI (ChatGPT 2025)

Compromised IoT devices can act as attack vectors, affecting larger networks such as corporate and governmental infrastructure. Experts advocate for secure-by-design principles due to this risk, emphasizing built-in encryption, robust authentication, and continuous patching (Bygrave, 2022).

Purpose, Scope, and Research Methodology

This study focuses on improving the security of IoT smart home devices by employing penetration testing techniques with Kali Linux, a commonly used ethical hacking platform, to test the security and integrity of Smart Home IoT Devices with a view to offering suggestions on improvements. The emphasis is on pinpointing prevalent vulnerabilities, replicating real-world attack scenarios, and suggesting effective countermeasures to enhance device resilience.

Scope of the Study:

a) Analyzing IoT communication protocols like HTTP, MQTT, CoAP, Zigbee, and their associated security flaws.

b) Performing penetration tests on smart home devices, such as security cameras and environmental sensors, utilizing Nmap, Wireshark, Metasploit, and Aircrack-ng.
c) Analyzing real-world IoT attack cases to assess their effects on smart homes.

Methodology of Research: A mixed-method approach integrates literature review, experimental testing, and case study analysis. Penetration tests occur in controlled settings utilizing various Kali Linux tools, following ethical hacking standards and privacy regulations (Thomas et al., 2019).

2 BACKGROUND

This chapter examines the technical background of smart home systems, following the identification of the importance of IoT and the cybersecurity challenges associated with these technologies. This document analyzes IoT communication and data protocols, identifies vulnerabilities, and discusses the importance of penetration testing for enhancing device security.

2.1 IoT's Arrival in Smart Homes

The emergence of IoT has significantly altered modern living by facilitating smart devices that improve daily life via automation, convenience, and efficiency. In residential environments, IoT facilitates the integration and remote management of various home systems, such as lighting, heating, air conditioning, security cameras, and entertainment units. According to Zaidan et al. (2018), these devices utilize wireless networks for communication and operation, forming an integrated and responsive smart home ecosystem. The ongoing advancements in artificial intelligence, wireless communication, and cloud computing are projected to significantly expand the smart home market, influencing the future of urban and suburban living.

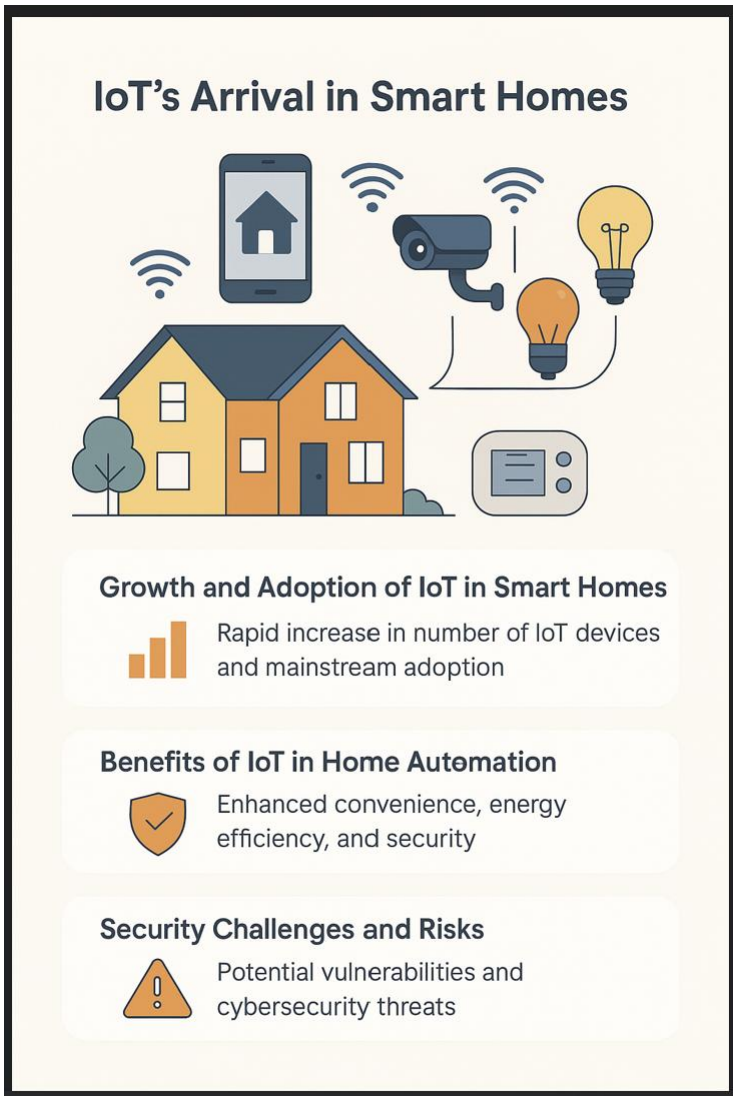


Figure 2. Figure 2. IoT's Arrival in Smart Homes. Image created with AI (ChatGPT 2025)

2.1.1 Growth and Adoption of IoT in Smart Homes

The worldwide growth of IoT technologies has accelerated adoption in the smart home industry. The IoT has evolved from early experimental phases to a fundamental aspect of contemporary home life. According to Papatsimouli et al. (2022), the number of IoT devices worldwide is projected to reach 50 billion by 2030, a dramatic increase from 22 billion in 2018. This exponential growth indicates both technological advancement and rising consumer demand for connected living environments. Leading technology firms, including Google, Amazon, and Apple, have made significant investments in smart home

ecosystems, providing products such as Google Home, Amazon Alexa, and Apple HomeKit. These platforms enable users to control their home environments seamlessly and intuitively via voice commands and mobile applications, thereby promoting the adoption and normalization of IoT in households.

2.1.2 Benefits of IoT in Home Automation

The incorporation of IoT within residential settings offers several distinct benefits. The main benefit is enhanced convenience, allowing homeowners to automate routine tasks such as thermostat adjustments, light control, and remote activation of appliances. Energy efficiency provides a notable benefit; intelligent systems improve power utilization by analyzing usage patterns and disabling inactive devices, thus fostering cost savings and environmental sustainability. Additionally, home security has improved due to IoT advancements, including motion detectors, smart locks, and surveillance systems that facilitate real-time monitoring (Magara & Zhou 2024). Moreover, personalization is crucial, as IoT devices can learn user preferences and behaviours, thus adjusting environments to fulfil specific needs. These features collectively improve comfort, efficiency, and security in residential living.

2.1.3 Security Challenges and Risks

The widespread adoption of IoT in smart homes poses significant security challenges, notwithstanding its benefits. The characteristics of interconnected systems suggest that each device, within a network, functions as a potential access point for cyber threats. A compromised device may facilitate access to the entire smart home system, leading to privacy breaches, unauthorized entry, and possible manipulation of critical home functions. Hewitt and Cunningham (2022) emphasize that the complexity and heterogeneity of smart home devices, often manufactured by various producers with inconsistent security standards, exacerbate these vulnerabilities. Securing smart homes necessitates a proactive approach involving the identification, monitoring, and mitigation of vulnerabilities.

Key elements in protecting these systems include strong encryption, frequent firmware updates, network segmentation, and user education. The lack of these measures may lead to a compromise in security and privacy in the context of smart living convenience.

2.2 Communication Protocols in Smart Homes

For IoT devices to be useful and fit for intended purpose, they need to communicate with themselves and also with their users. This communication can happen over long distances through the versatility of the internet networks and also through protocols that utilize their proximity to each other. The rise in adoption of IoT systems necessitates an understanding of the communication protocols employed by these devices to identify potential vulnerabilities (ENISA 2021; MQTT.org, n.d.; Zigbee Alliance, 2024). This section briefly describes key wireless protocols and their related security concerns.

2.2.1 Hypertext Transfer Protocol (HTTP/HTTPS)

HTTP is a key standard that simplifies communication between web apps and Internet of Things devices. Smart home appliances connect to cloud services via HTTP but HTTP is susceptible to man-in-the-middle attacks and interception due to its transmission of data in plain text (Conti et al., 2016). To address this vulnerability, Secure HTTP (HTTPS) encrypts data during transmission with Transport Layer Security (TLS) (McKay & Cooper, 2017).

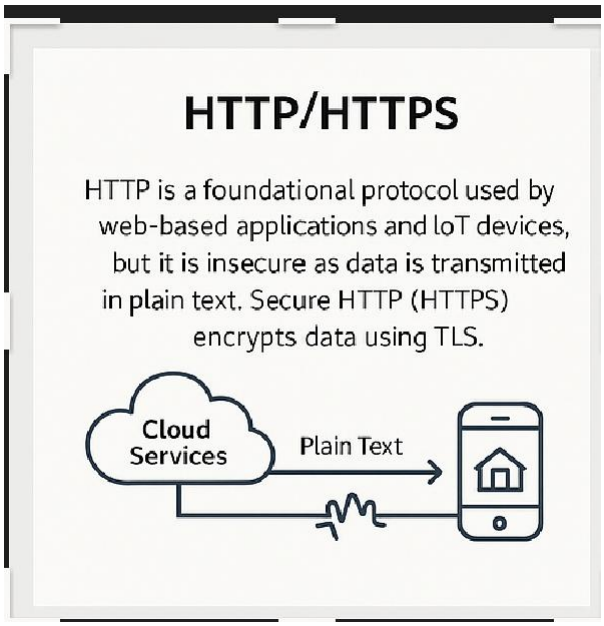


Figure 3. Figure 3. Plain Vulnerability in HTTP and TLS Encryption in HTTPS. Accessed from Mozilla Web docs

2.2.2 Long Range Wide Area Network (LoraWAN)

LoRaWAN is ideal for smart homes requiring broad connectivity and minimal energy use, as it is designed for low-power, long-range IoT communication (Anand et al., 2024). Smart metering applications and environmental sensors utilize this technology effectively. Replay attacks and key leakage may occur due to inadequate implementation, even with end-to-end encryption and key management (Dechand et al., 2019).

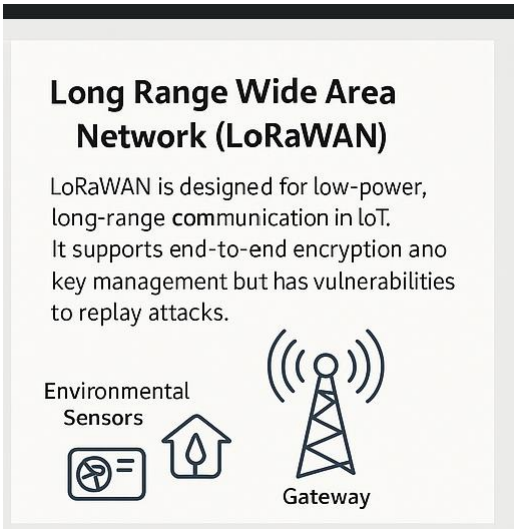


Figure 4. Figure 4. LoRaWAN Communication in IoT and it's Replay Attack Vulnerability. Adapted from public LoRaWAN specification

2.2.3 Bluetooth and Bluetooth Low Energy (BLE)

Bluetooth is a short-range wireless communication protocol frequently used in wearables, speakers, and smart locks. It enables devices to communicate over short distances, making it a crucial component in most IoT devices. Bluetooth Low Energy (BLE) is an improvement on traditional Bluetooth technology as it enhances energy efficiency but it is still vulnerable to attacks like eavesdropping and unauthorized pairing. To help improve Bluetooth security, it is recommended to Implement strong encryption and the secure pairing of systems (Barua et al., 2022).

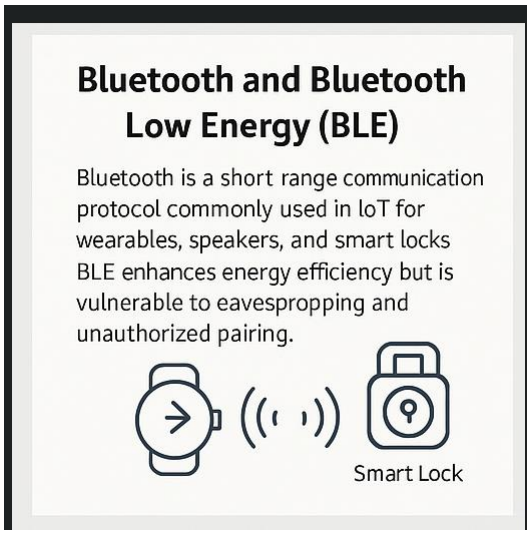


Figure 5. Figure 5. Bluetooth Low Energy Communication and Associated Vulnerabilities. Adapted from public Bluetooth documentation and IoT security studies

2.2.4 Zigbee

ZigBee is a low-power mesh networking protocol used in smart lighting, sensors, and thermostats. For its security, it employs AES-128 encryption but this has been criticized for weaknesses in key exchange and authentication, which can lead to spoofing and replay attacks (Dragomir et al., 2016.)

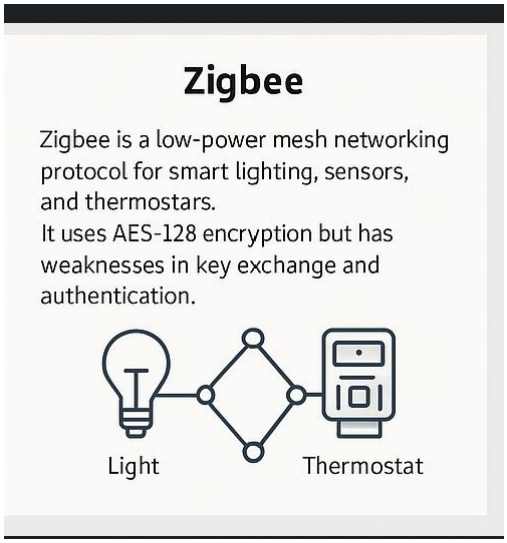


Figure 6. Figure 6. Zigbee Mesh Networking and Its Security Limitations. Adapted from Zigbee alliance association.

2.3 IoT Data Protocols and Associated Security Concerns

Smart home devices utilize specific data exchange protocols alongside communication protocols to facilitate information transmission between components. However, every protocol presents security trade-offs that must be considered to ensure robust smart home security (Asonze et al., 2024).

The process of securing IoT data is not as simple as selecting a protocol that is "secure" in theory; rather, it is dependent on a comprehensive and multi-layered approach to the construction, deployment, and maintenance of the lifecycle. To mitigate the pervasive and ever-evolving security risks that exist within the vast and interconnected world of the IoT systems, comprehensive strategies need to include robust authentication and authorization, strong encryption, secure key management, regular software updates, and continuous monitoring. These protocols are discussed in this section.

2.3.1 Constrained Application Protocol (CoAP)

CoAP is a lightweight protocol designed for low-power devices that utilize UDP for communication. CoAP demonstrates efficiency; however, it is susceptible to spoofing and does not incorporate inherent encryption mechanisms. Implementing Datagram Transport Layer Security (DTLS) can effectively reduce these risks (Maleh & Ezzati, 2016).

2.3.2 Advanced Message Queuing Protocol (AMQP)

AMQP is a middleware protocol that ensures reliable message delivery between devices (Al-Masri et al., 2020). Although it supports authentication and encryption, misconfigured systems may still be exposed to unauthorized access and data modification. Communication based on AMQP can be secured through the implementation of Transport Layer Security (TLS), as advised by the OASIS AMQP specification (Adiwal et al., 2024).

2.3.3 Message Queuing Telemetry Transport (MQTT)

MQTT is a commonly utilized protocol for efficient and lightweight messaging. The publish-subscribe model facilitates communication; however, it also presents risks, including denial-of-service (DoS) attacks and unauthorized access to messages. Implementing TLS and robust authentication protocols significantly improves security (Hintaw et al., 2023).

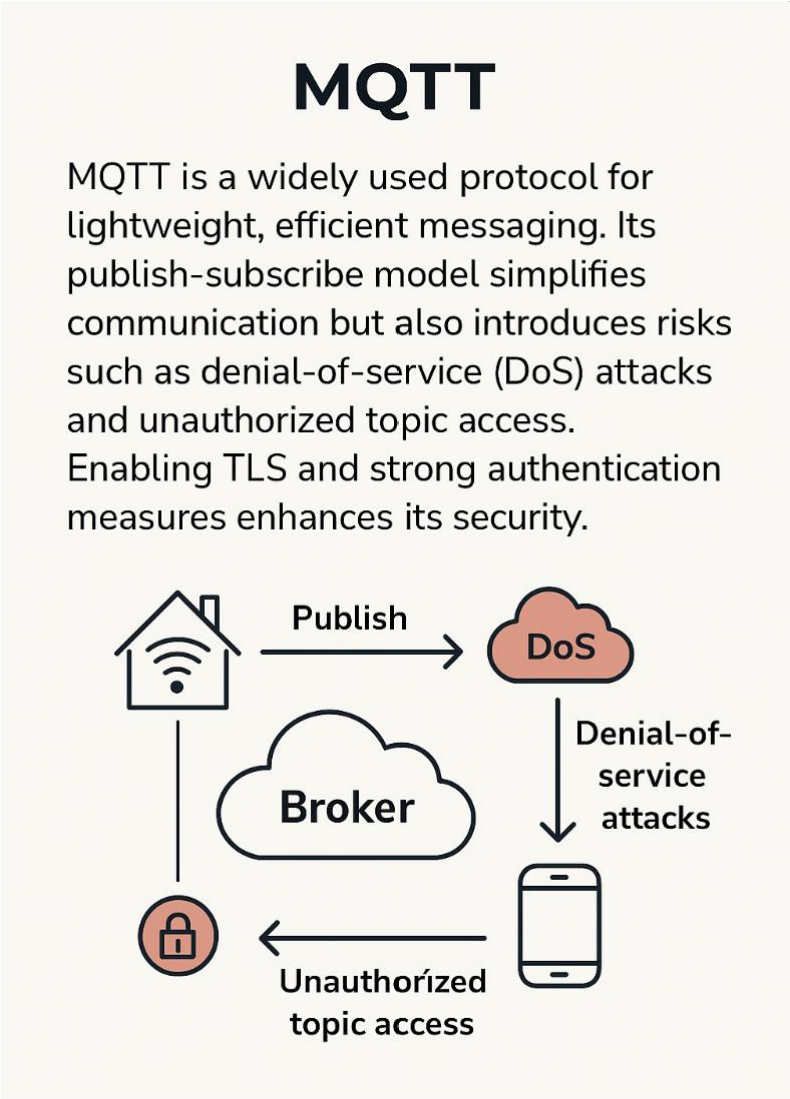


Figure 7. Figure 7. MQTT Publish-Subscribe Communication and Its Security Vulnerabilities. Adapted from public MQTT documentation

2.3.4 Extensible Messaging and Presence Protocol (XMPP)

XMPP was initially designed for instant messaging; however, it has since been extensively adopted for real-time communication in IoT devices. XMPP provides authentication and encryption; however, improper configuration may allow attackers to access messages or private data. XMPP-based smart home products exhibit enhanced safety when employing encryption and secure authentication (Iivari et al., 2014).

2.4 Common Security Issues in IoT Devices

Discussions regarding protocols indicate that smart homes continue to face several security challenges:

- **Weak Credentials:** such as default logins, are susceptible to brute-force attacks (Mansfield-Devine, 2021)
- **Unencrypted Data:** can be susceptible to Man-in-the-Middle (MitM) attacks when messages are transmitted in plaintext (Luna, 2020)
- **Outdated Software:** Devices face increased risk when updates are infrequent or delayed (Malhotra, 2021).
- **Protocol Vulnerabilities:** Abed and Anupam (2023) indicate that Wi-Fi, Zigbee, and Bluetooth possess security vulnerabilities that allow unauthorized individuals to eavesdrop on communications.
- **Lack of Standards:** Inconsistent security practices across vendors undermine overall security (Aly et al., 2019).

2.5 The Role of Penetration Testing

Penetration testing, which is widely known as pen testing, is a systematic process that simulates different attack scenarios to evaluate the security of an IT system (Cybermaxx, 2023). Penetration testing, as a critical method for identifying and mitigating threats, enables researchers and security experts to identify and rectify vulnerabilities that may provide avenue for exploitation by hackers. The pen testing process consists of several phases: reconnaissance, which involves gathering information about the target; scanning and enumeration, aimed at identifying potential entry points; exploitation, where

attempts are made to gain unauthorized access; post-exploitation analysis, which assesses the extent of compromise; and reporting, which details findings and recommendations (Sarker et al., 2023).

Overall, penetration testing contributes in ensuring the robustness of device firmware, network configurations, and communication methods by simulating real-world attacks. Developers can enhance the security of their systems by implementing these strategies (Weir et al., 2023).

2.6 Conclusion

Various communication and data methods are employed in smart home systems. Each option presents distinct advantages and disadvantages regarding security. Encryption and authentication can mitigate various risks; however, the complexity and constant evolution of the IoT necessitates continuous research and secure-by-design methodologies.

The subsequent chapters will examine how testing conducted with Kali Linux can identify and rectify vulnerabilities in actual smart home devices.

3 THREAT MODELLING AND ETHICAL CONSIDERATIONS

Smart homes enabled by Internet of Things devices offer improved convenience and automation (Taiwo et al., 2022). Smart homes allow homeowners to control lighting and climate settings through smartphones and receive real-time alerts from advanced security systems. These systems are vulnerable to various cybersecurity threats due to their growing reliance on wireless and cloud-based infrastructures. The previous chapters demonstrate that these vulnerabilities significantly affect network integrity, device performance, and data confidentiality.

This chapter analyzes the cybersecurity of smart homes by classifying potential threats based on the Confidentiality, Integrity, and Availability (CIA) triad, a recognized framework in cybersecurity threat modelling (Shafik et al., 2023). This chapter examines the importance of ethical penetration testing in detecting vulnerabilities, as well as the legal and ethical responsibilities that researchers and security professionals must adhere to during these evaluations.

3.1 Threat Modelling Using the CIA Triad

Threat modelling is a foundational process in cybersecurity to systematically identify, categorise for easy mitigation of potential threats to information systems. The CIA triad which encompasses confidentiality, integrity, and availability threats provides this structured framework for categorizing cybersecurity issues in IoT devices. This approach provides a framework for easy identification, categorisation and understanding of these vulnerabilities in smart home networks, thereby offering substantial support.

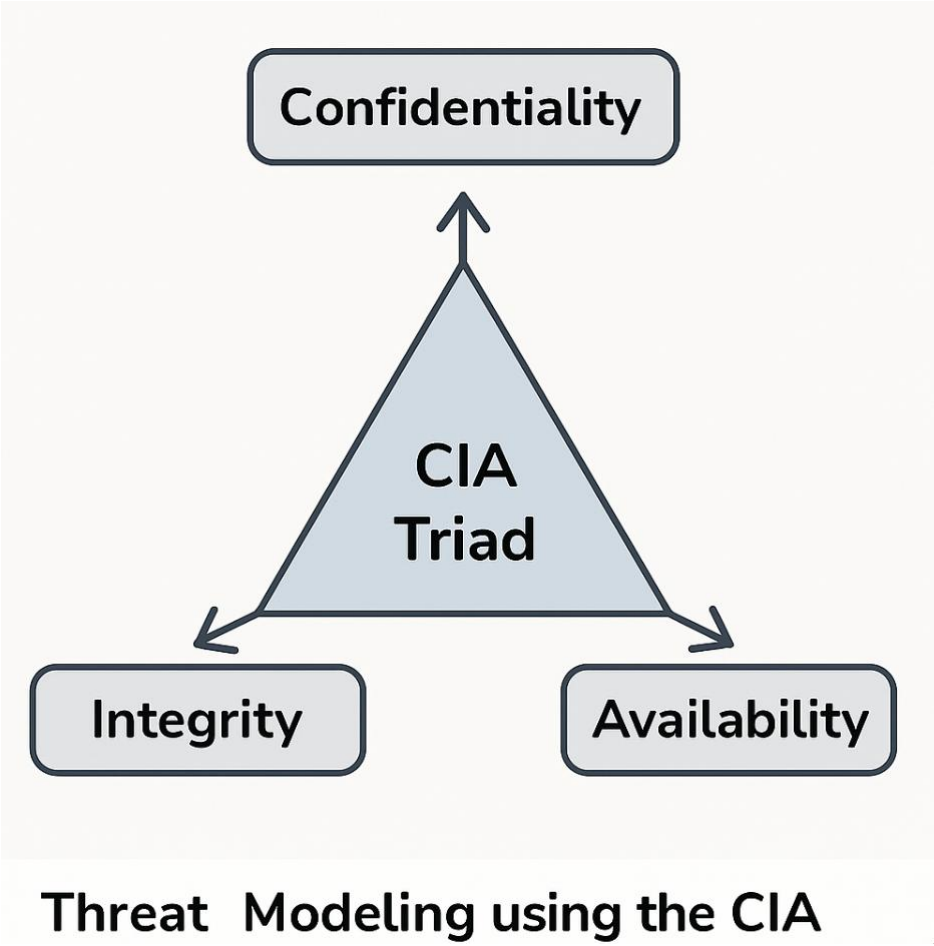


Figure 8. Figure 8. Threat Modelling using the CIA Triad. Adapted from NIST SP 800-12 Rev (NIST, 2003)

3.1.1 Confidentiality Threats: Data Leaks & Unauthorized Access

Generally, access to essential information is restricted to authorized personnel, thereby maintaining confidentiality. However, with the advance of technology, a variety of Internet of Things devices now handle personal data, rendering them susceptible to fraudulent activities.

Data Leaks: Devices that transmit data over unencrypted channels are susceptible to eavesdropping, which may result in data breaches. Adversaries can employ tools such as Wireshark to access sensitive information, including credentials and usage patterns (Soepeno 2023).

Unauthorized Access: Brute-force and credential stuffing attacks frequently take advantage of weak default credentials present in the system. Antonakakis et al.

(2017) demonstrate the Mirai botnet as a significant example of widespread device vulnerability stemming from inadequate password security.

Risk Mitigation Strategies: Mohammed et al. (2023) identify essential security measures such as implementing least-privilege access protocols, utilizing multi-factor authentication, and enforcing AES-256 encryption standards.

3.1.2 Integrity Threats: Code Injection and Manipulation

Safeguarding data against unauthorized changes is essential for maintaining integrity. Devices within the Internet of Things systems that transmit and receive data from sensors or control systems can exhibit vulnerability to tampering.

Tampering: This involves the ability of attackers to intercept and modify communications between servers and devices. Tu et al. (2019) demonstrate that modifying data on a thermostat can cause issues in an HVAC system.

Code Injection: Malicious entities exploit software vulnerabilities to gain control of systems such as smart locks or surveillance devices by introducing harmful code (Van & O'Flynn, 2021).

Risk Mitigation Strategies: Zandberg et al. (2019) posit that implementing input validation, code signing, and secure development practices greatly reduces the risk of these attacks.

3.1.3 Availability Threats: Denial of Service & Resource Exhaustion

A cardinal reason for the widespread acceptance of advances in information technology is the ability it provides for increased access by people and devices. Systems and data provide necessary accessibility. However, due to limited resource availability, Internet of Things devices are highly vulnerable to disturbances that can restrict or inhibit this access.

Denial of Service (DoS) Attacks: A DoS attack occurs when an attacker overwhelms a network or device with excessive traffic, depleting its resources

and rendering it inoperable. The Dyn DNS attack illustrates the hijacking of Internet of Things devices, resulting in the disruption of internet services in multiple regions (Xenofontos et al., 2021).

Resource Exhaustion: A considerable number of Internet of Things devices display limitations in processing power and memory, which may lead to resource exhaustion. Attackers leverage this vulnerability by sending continuous queries that inundate devices. A perpetrator can disable a smart camera by making numerous connection attempts (Sadhu et al., 2022).

Risk Mitigation Strategies: Chaudhary and Kumar (2019) argue that system resilience can be improved through mitigation measures such as traffic filtering, load balancing, anomaly detection, and network segmentation.

3.2 Ethical Considerations in IoT Penetration Testing

Penetration testing is essential for identifying security vulnerabilities in IoT devices. The design simulates authentic hacking, necessitating that individuals adhere to stringent legal and ethical standards.

3.2.1 Informed Consent and Privacy Protection

Fake hacks can potentially compromise or expose private user data. Prior to testing, ethical testers must obtain explicit written consent from the device owners or relevant stakeholders (Wilhelm, 2021). External testing may violate privacy regulations such as GDPR and state hacking statutes

3.2.2 Responsible Disclosure Practices

Security experts must exercise caution when discussing vulnerabilities in systems. Developers of gadgets and their software must be informed of vulnerabilities to address them prior to public release. Malicious individuals might exploit careless sharing for their advantage (Ding et al., 2019).

3.2.3 Legal and Regulatory Compliance

Entering without authorization, even for research purposes, could constitute a legal violation. Penetration testers must adhere to security regulations established at foreign, national, and internal levels. Testing is conducted in compliance with guidelines such as ISO/IEC 27001 and legal requirements, ensuring adherence to the law (Qusef & Alkilani, 2022).

3.3 Summary

This chapter further examined cybersecurity challenges associated with IoT-enabled smart homes and the increasing risks posed by their reliance on wireless and cloud infrastructures. Using the CIA triad for categorization, it provides a structured framework for modelling these IoT threats and devising ways of testing for them.

The chapter also emphasized the importance of adhering to ethical guidelines during penetration testing. Ethical testers must ensure that simulations of real-world attacks are conducted responsibly and within regulatory frameworks to protect user privacy and system integrity.

The next chapter will introduce Kali Linux and its associated tools that will be utilized for this ethical penetration testing on select smart home devices.

4 METHODOLOGY

This chapter describes the methodology used for penetration testing the Airam smart temperature and humidity sensor, an IoT device within a controlled local network, and a generic Smart Home Sensor (with a temperature and humidity detector) bought online. Testing took place in the Kali Linux environment to simulate attack vectors, identify vulnerabilities, and assess the device's security configuration. The approach ensures adherence to legal requirements by fusing ethical hacking concepts with actual situations.

4.1 Tools and Testing Environment

The tools and components utilized for the penetration tests in this project are as follows:

- **Target IoT Devices**: Airam SmartHome device (Device 2) and a generic Smart Home device (Device 1). Both are environmental sensors commonly utilized in smart home environments
- **Penetration Testing Platform**: Kali Linux operating within a virtual machine, featuring tools such as Nmap, Hydra, Metasploit, Wireshark, and Tcpdump.
- **Network Configuration**: A private home network established with the tester and devices operating on the same subnet. A domestic Buffalo AirStation router was the network device.
- **Nmap**: For port scanning, OS fingerprinting, traceroute, and service enumeration.
- **Hydra**: For brute-force login testing using dictionaries such as rockyou.txt.
- **Metasploit Framework**: For vulnerability assessment and preparation of non-destructive exploitation.
- **Wireshark/Tcpdump**: For passive traffic monitoring and packet capture.

Test Bed Validation Set-up.

The first task was the establishment of a controlled testbed environment. This was to ensure that all testing activities were carried out, away from production systems, and were conducted ethically (Siboni et al., 2018).

Two initial commands were run in order to verify this test environment:

1. The command “tcpdump -h” provides a list of available options for utilizing the robust packet capture tool known as tcpdump. This step is essential for configuring the attacker machine in preparation for packet sniffing.
2. The command 'ip a' displays the network interfaces that have been configured. The primary external interface (eth0) is assigned the IP address 10.0.2.15, which serves to identify the Kali Linux attacker machine throughout the scanning and exploitation tests.

The collection of this information confirms that the penetration test environment has been properly established, featuring a working IP stack and the ability to sniff traffic. This setup serves as the essential groundwork for the following phases of reconnaissance and analysis.

This screenshot below illustrates the application of these two essential setup commands.

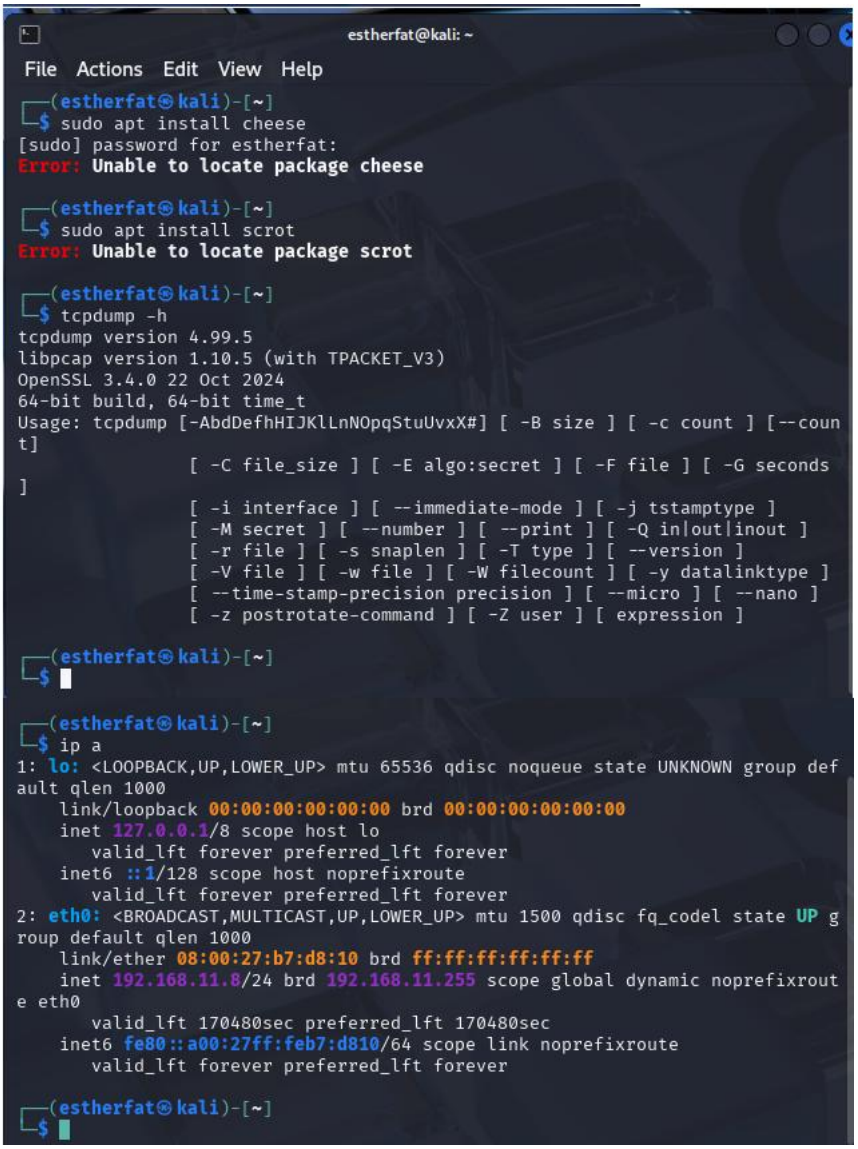


Figure 9. Figure 9. tcpdump-h and "ip a" Configuration Output on Kali Linux (Device 2).

4.2 Penetration Testing Workflow

The testing followed the listed phases:

1. Reconnaissance
2. Scanning and Enumeration
3. OS Fingerprinting and Traceroute
4. Vulnerability Assessment
5. Exploitation (non-destructive)
6. Traffic Monitoring and Packer Analysis

7. Reporting

4.2.1 Reconnaissance with Nmap

The reconnaissance phase focused on identifying live hosts, scan for open ports, and detect running services and different Nmap scans were used. These were done to identify the target device, scan for open ports, and detect running services. A SYN stealth scan was employed to minimize detection by the device while collecting this information. This included both SYN scanning and version detection to gather detailed insights into the services exposed on the device. Different Nmap commands were employed for this reconnaissance task.

Device 1 (78.27.71.99)

To initiate reconnaissance, a comprehensive Nmap scan was executed using the following command: Nmap -p- -sV -A 78.27.71.99 as shown in the figure below

Figure 10. Initial Nmap Output Scan Showing Initialization and Port Discovery (Device 1)

To achieve an Extended TCP and UDP Reconnaissance Scan, the command sudo nmap -sS -sV -p- 78.27.71.99 was repeated on the same device and the screenshot is below

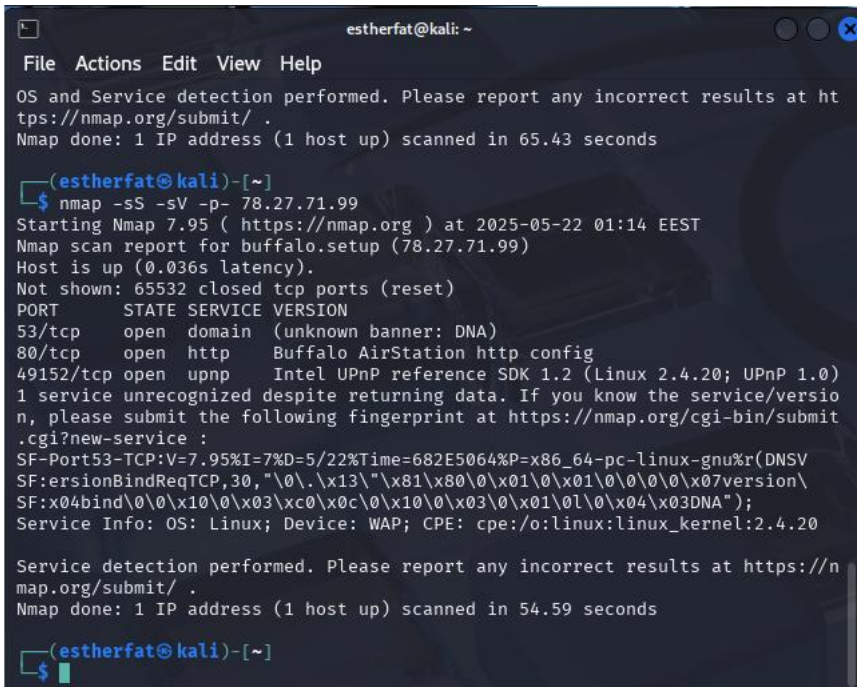


Figure 11. Extended TCP and UDP Reconnaissance Scan (Device 1)

This second run gives an advanced reconnaissance of the device as SYN scan, Service/Version detection and complete scan of all ports on the device were carried out. The following details were revealed:

TABLE 1. Details from Advanced Reconnaissance Scan of Device 1.

PORT	STATE	SERVICE	VERSION/DESCRIPTION
53	OPEN	Domain	Shows a detailed DNS fingerprint
4512	OPEN	HTTP	Buffalo AirStation config, returns HTTP 401 Unauthorized
49152	OPEN	UPnP	Intel UPnP SDK 1.2 on Linux 2.4.20
1723	OPEN	PPTP	Point-to-Point Tunneling Protocol (PPTP)

Device 2 (85.131.120.225)

For this device a basic port discovery was carried out on device 2 using the command: sudo nmap 85.131.120.225

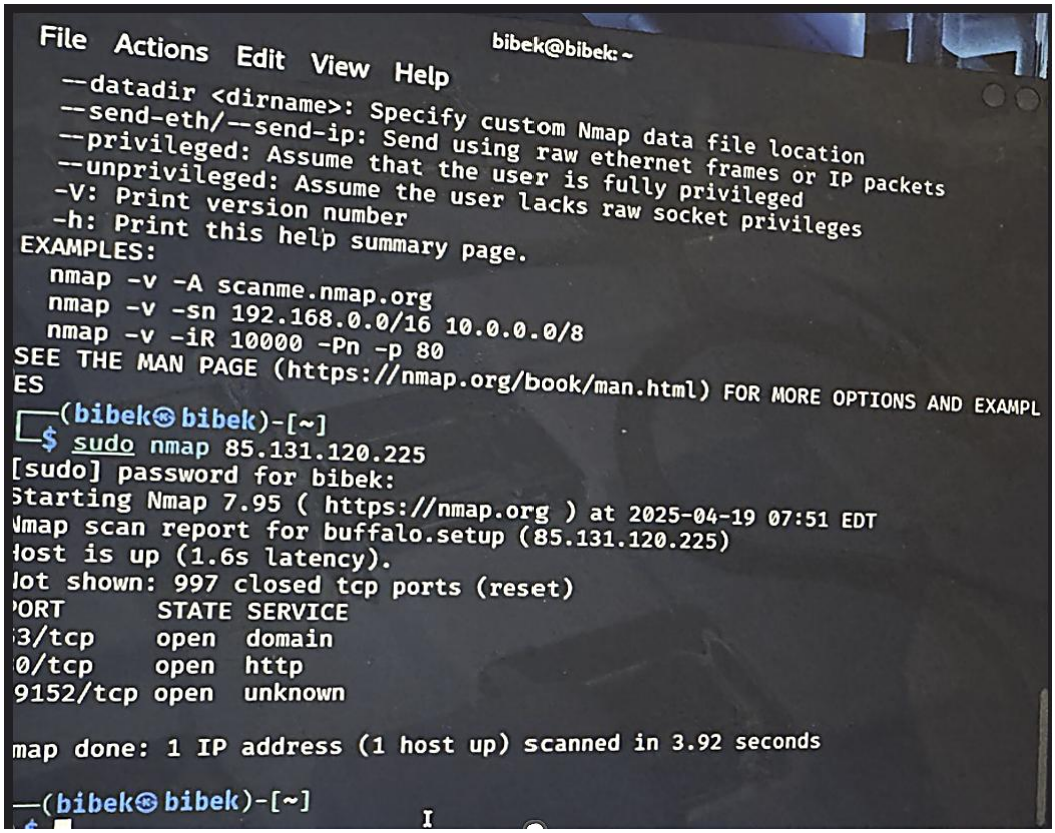


Figure 12. Basic Port Discovery with Nmap (Device 2).

This scan attempts to identify which of the most common 1000 TCP ports are open on the target device. The output indicates that 997 out of the top 1000 scanned TCP ports were closed, meaning no service is listening on those. The following open ports were detected:

TABLE 2. Details from Reconnaissance Scan of Device 2.

PORT	STATE	SERVICE	VERSION/DESCRIPTION
53	OPEN	Domain	Indicates a DNS service is running
80	OPEN	HTTP	Standard web service; may host a web UI

9152	OPEN	Unknown	Port open but service is unrecognized by Nmap

Continuing with Device 2 to get more detailed information about the device including service version information, an advanced Reconnaissance + Service & OS Detection was carried out to scan all ports on the device. To achieve this, a SYN scan combined with service version detection was conducted using the command: sudo nmap -sS -sV -p- 85.131.120.225. The screenshot below shows the result of the scan.

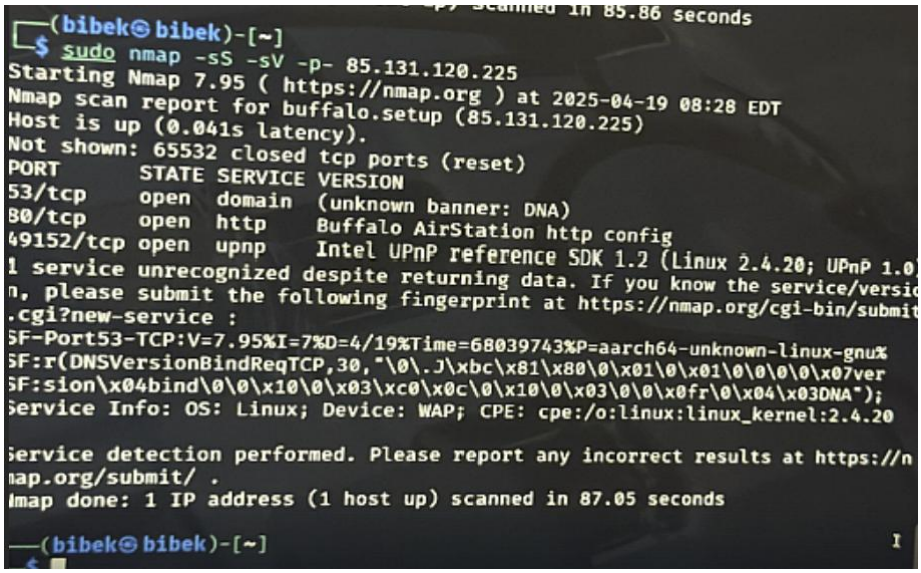


Figure 13. Nmap Output Showing Open Port Discovery (Device 2).

This scan revealed open ports such as 80/tcp (HTTP), 53/tcp (DNS), and 49152/tcp (UPnP). These ports indicated that the device hosted a web interface, possibly a DNS service, and used universal plug-and-play protocols. Find below table 2 showing this information.

TABLE 3. Details from Open Port Discovery from Device 2.

PORT	STATE	SERVICE	VERSION/DESCRIPTION
53/tcp	OPEN	Domain	Unknown (banner:DNA)

80/tcp	OPEN	HTTP	Buffalo Airstation HTTP config
49152/tcp	OPEN	UPnP	Intel Upnp SDK 1.2 (Linux 2.4.20; Upnp 1.0

4.2.2 Service Enumeration Scan

This is the process of probing open network ports to identify the specific services running on them, including application names, version numbers, and configuration details. The command used in the Nmap tool in the previous subsection also achieve service enumeration due to the inclusion of -sV flag in it.

Device 1 (78.27.71.99)

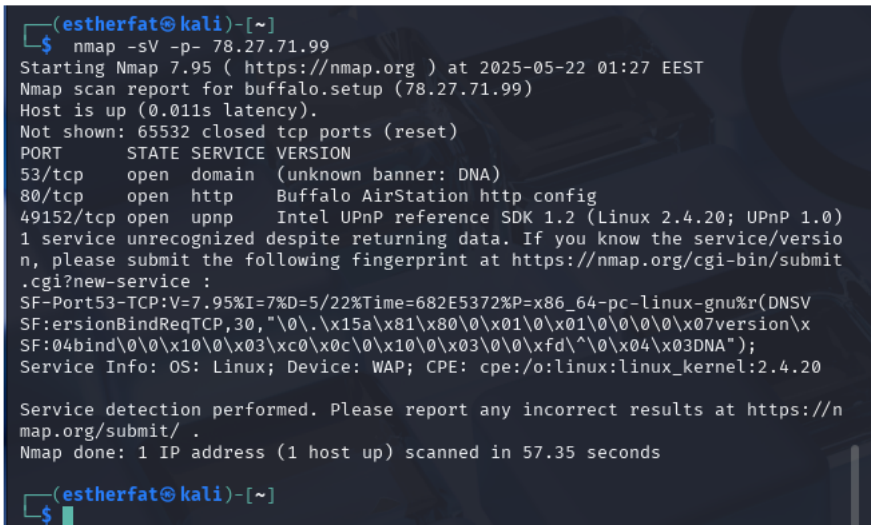


Figure 14. Nmap Service Enumeration Scan (Device 1).

The device was identified as a general-purpose Linux-based host, likely embedded. This level of service exposure is typical in consumer-grade smart home devices and underscores the importance of evaluating access points systematically. Table 4 below details the findings.

TABLE 4. Details from Enumeration Scan of Device 1.

PORT	STATE	SERVICE	VERSION/DESCRIPTION
53/tcp	OPEN	DNS	Open DNS service with NSID metadata and BIND version info exposed
80/tcp	OPEN	HTTP	Web interface prompting for HTTP Basic Authentication
49152/tcp	OPEN	UPnP	Intel UPnP SDK 1.2 on Linux 2.4.20; known to have vulnerabilities if unpatched
55171/tcp	OPEN	tcpwrapped	Port responded but access restricted or filtered (TCP wrapper in use)

4.2.3 OS Fingerprinting and Traceroute

Device 1

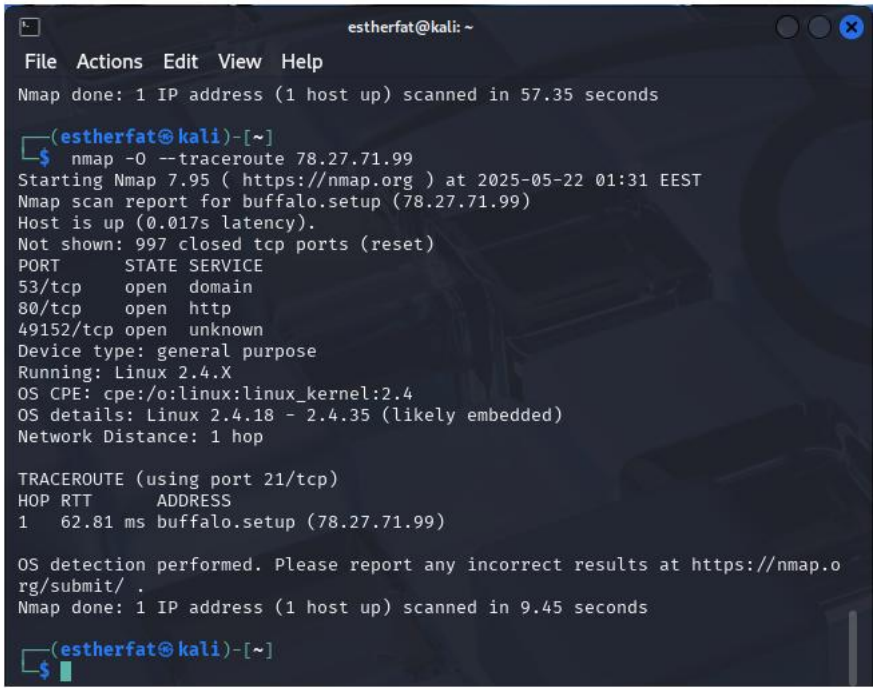


Figure 15. Nmap OS Fingerprinting and Traceroute Output (Device 1)

This screenshot above shows detailed results from Nmap's OS detection, traceroute, and NSE (Nmap Scripting Engine) post-scan activities for Device 1. An aggressive Nmap scan was conducted on the device with IP address 78.27.71.99 to perform OS detection, service identification, and traceroute analysis using TCP port 143. The following results were obtained:

• OS: Linux 2.4.18 to 2.4.35

• Device type: WAP

• Traceroute: 1 hop, indicating local or routed proximity

• NSE script analysis initiated post-san

4.2.4 Vulnerability Assessment

To test password strength and access control, a Hydra brute-force attack was launched against HTTP service on Device 1 using a standard wordlist. Command: hydra-l admin -p /usr/share/wordlists/rockyou.txt 78.27.71.99 http-get

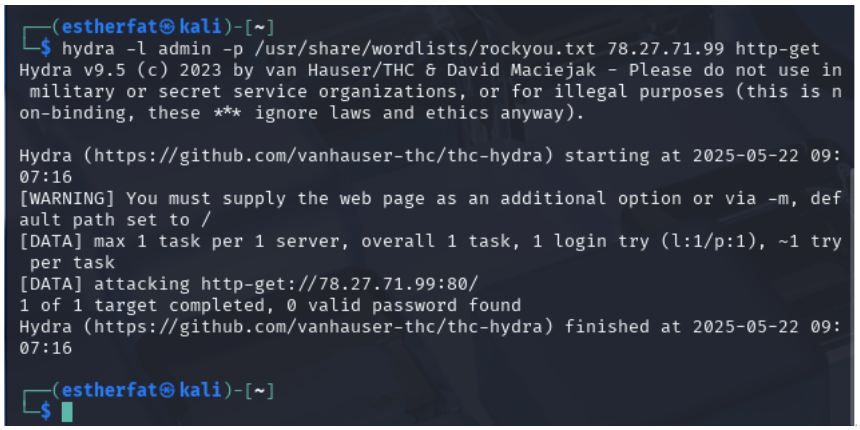


Figure 16. Brute-force Login Test on Device 1.

This was a basic attack but as can be seen from the screenshot, none of the passwords in the rockyou list matched the admin account at the default web page path /.

4.2.5 Exploitation (non-destructive)

In this phase, the Metasploit tool in Kali Linux was investigated and the different modules for targeting vulnerabilities were explored. Also, different non-destructive exploitation procedures were carried out on both devices.

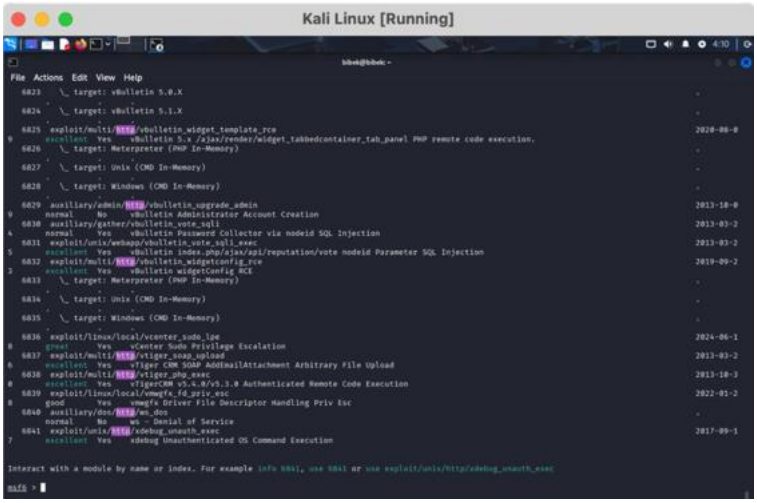


Figure 17. Metasploit Framework Search Action.

Figure 17 displays the output of a Metasploit search query, listing various HTTP-based remote code execution modules. Modules searched included:

• HTTP-based RCE (e.g., vbulletin_widget_template_rce, xdebug_unauth_exec)

• UPnP vulnerabilities (e.g., miniupnpd, Realtek SDK)

• TCP payloads for reverse/bind shells across architectures

The presence of the unix/http/xdebug_unauth_exec module confirms that Metasploit can be used to simulate potential exploitation attempts against misconfigured or outdated web services. This step is typically a part of vulnerability research and module selection before exploitation attempts.

UPnP Vulnerabilities

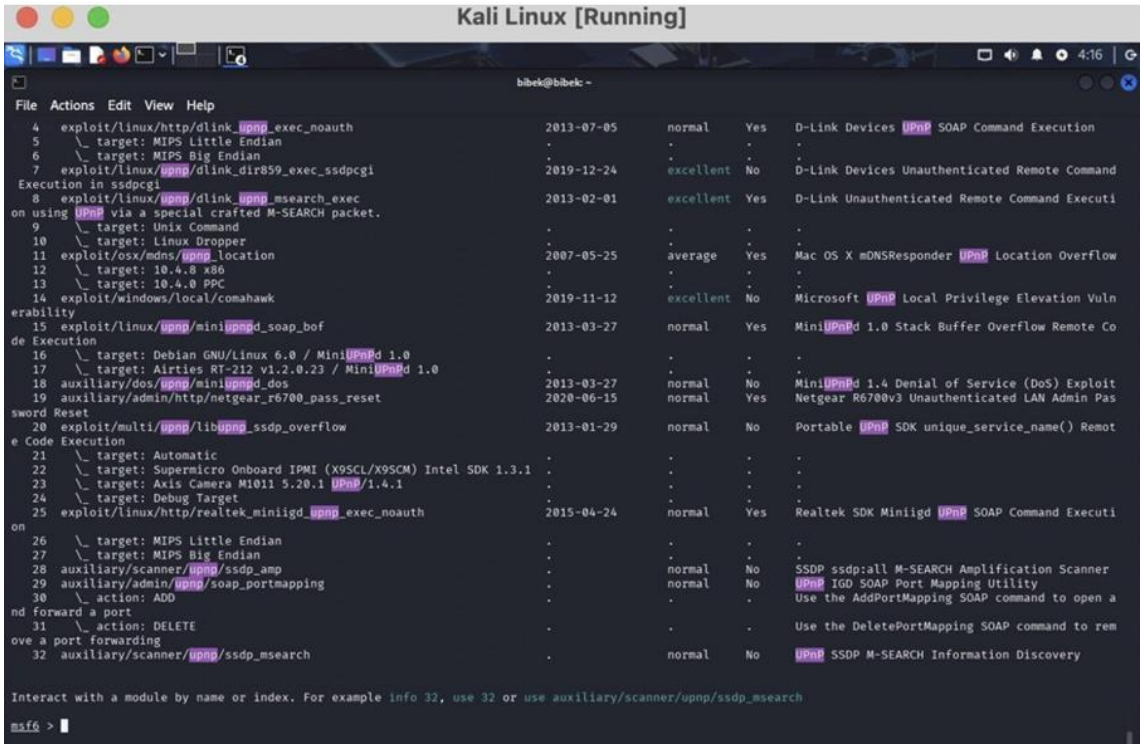


Figure 18. Examination of Metasploit Modules Targeting UPnP Vulnerabilities.

Figure 18 presents a list of Metasploit modules specifically related to UPnP (Universal Plug and Play) vulnerabilities. It includes exploits targeting D-Link devices, Realtek SDK, MiniUPnPd, and various scanner utilities. These modules range from remote command execution to denial-of-service and information discovery, illustrating the range of attack vectors available via UPnP misconfigurations. This phase of the methodology aimed to identify potential vulnerabilities associated with the UPnP service that could be used target IoT devices.

TCP in Metasploit

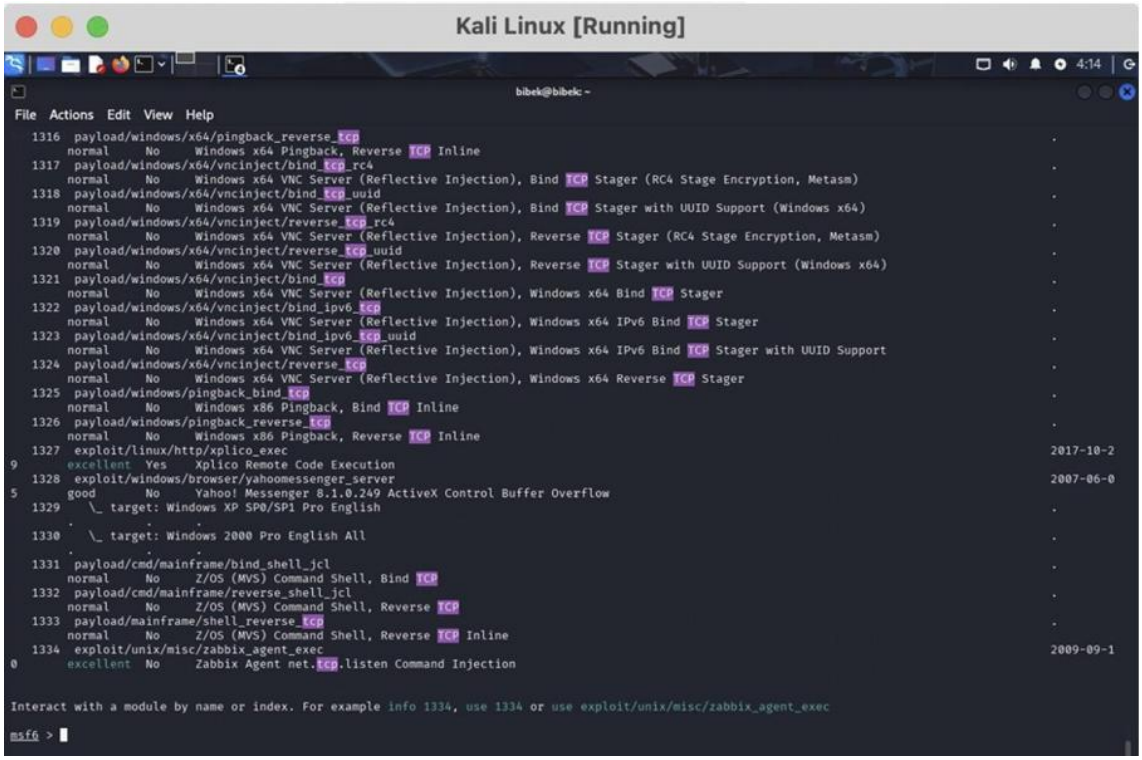
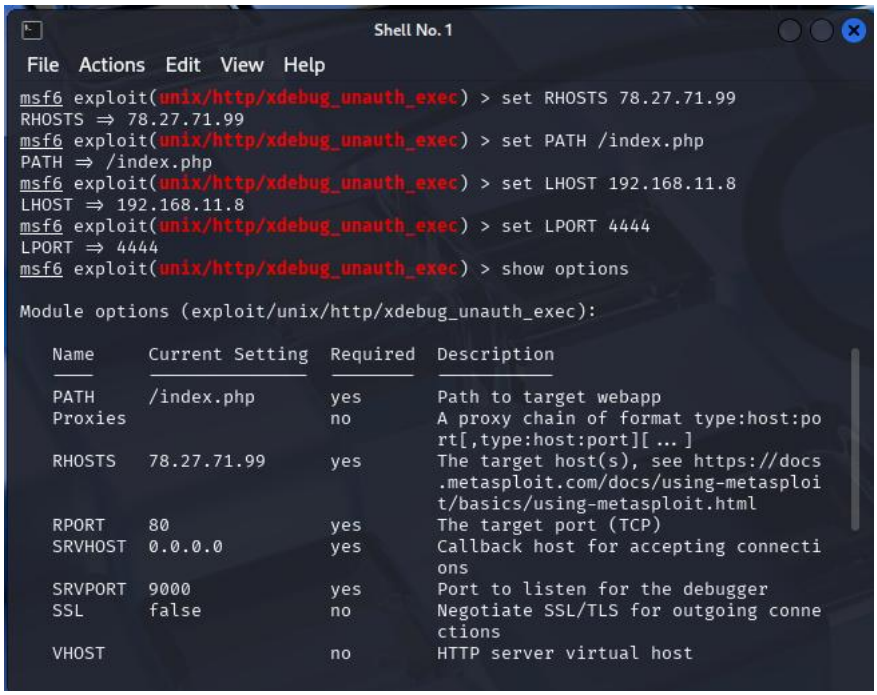


Figure 19. Listing of TCP-based Payloads and Exploits in Metasploit.

Figure 19 shows a comprehensive list of TCP-based payloads and exploits available within the Metasploit Framework. The package encompasses payloads for various architectures, including x64, x86, and mainframe systems such as Z/OS. The modules encompass various functions, including reverse TCP shells, bind shells, and targeted exploits such as zabbix_agent_exec. This screen illustrates the variety of attack vectors accessible through TCP communication protocols, highlighting how Metasploit facilitates testing across diverse systems and configurations.

Attempts to Exploit the Devices

The next step was to attempt exploitation of the devices using Metasploit



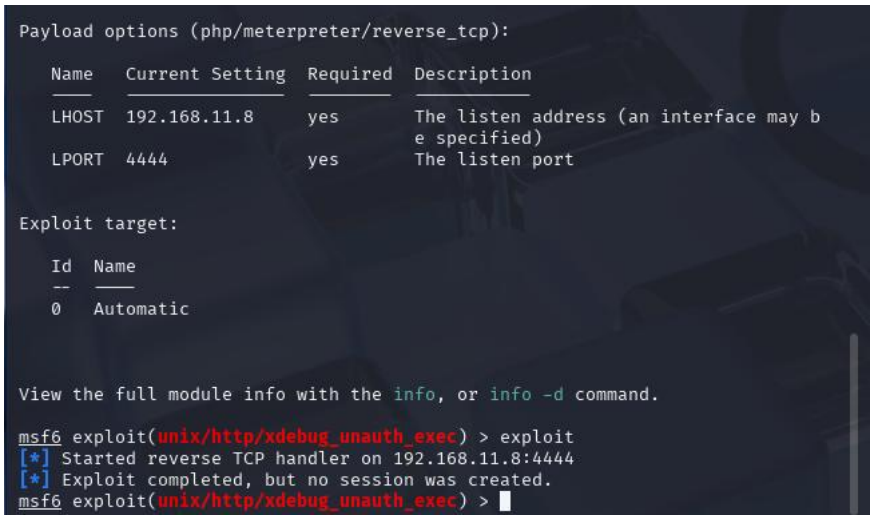


Figure 20. 2 Screenshots showing Xdebug Exploit Attempt (Device 1)

This screenshot illustrates the configuration phase of the Metasploit unix/http/xdebug_unauth_exec exploit module. The terminal window shows the setup of the payload php/meterpreter/reverse_tcp, along with the specification of key module options such as RHOSTS, RPORT, SRVHOST, LHOST, and LPORT. These parameters define the target host, the remote web service port, and the attacker's callback address and port for receiving a reverse shell.

The module is set to attack a potentially vulnerable Xdebug PHP endpoint (presumed located at /index.php) over HTTP on port 80. This step was part of

preparing a remote code execution attempt. Although the exploit was executed successfully and a reverse TCP handler was initiated on the attacker's end, no session was established, indicating that either the target was not vulnerable to this exploit or a connection was blocked.

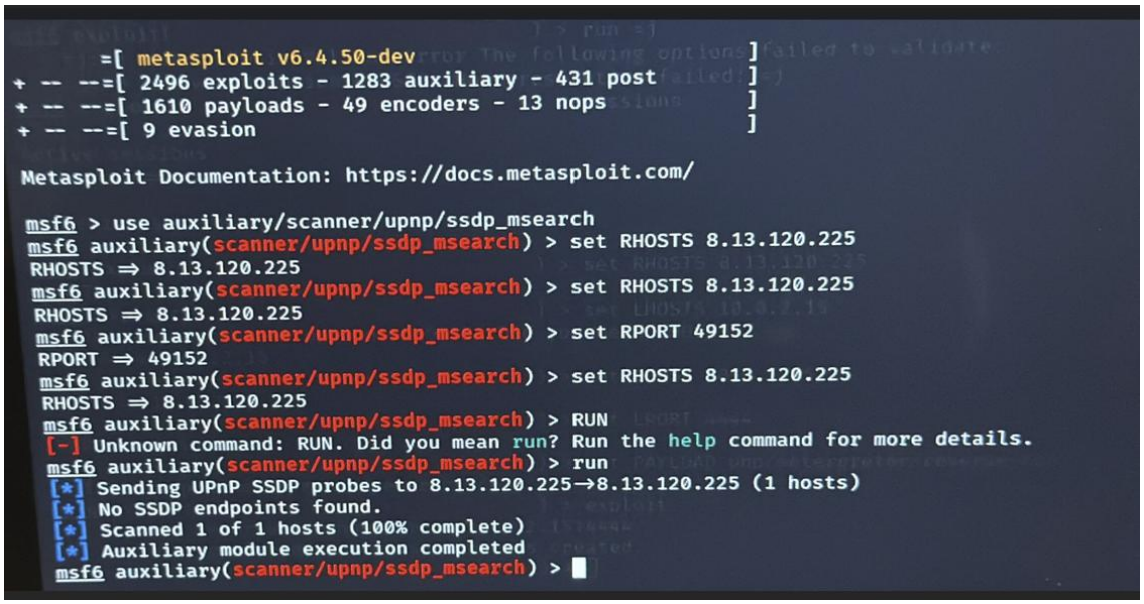


Figure 21. Metasploit SSDP UPnP Scan Attempt (Device 2).

Figure 21 shows the use of the Metasploit Framework to scan the target device for UPnP SSDP (Simple Service Discovery Protocol) endpoints using the auxiliary/scanner/upnp/ssdp_msearch module. The goal of the scan was to identify devices exposing UPnP services, which are commonly used in IoT smart home environments.

The module was set up with the designated IP address and port (49152) and executed via the run command. The results indicate that the target was scanned; however, no SSDP endpoints were identified. This indicates that the device may not be utilizing UPnP, may not be accessible on the network, or may be safeguarded by a firewall.

Although no positive findings were obtained, this test is crucial for determining the device's susceptibility to UPnP-based enumeration or attacks, which is a recognized vulnerability in numerous IoT smart home devices.

4.2.6 Traffic Monitoring and Packet Analysis

Packet Capture Attempt with Tcpdump

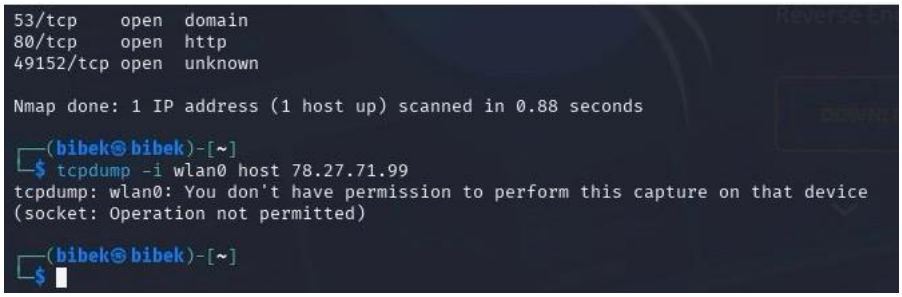


Figure 22. Tcpdump Attempt (Device 1).

This is a network sniffing test using tcpdump, a common tool for capturing and analyzing network packets. The intention was to:

• Capture packets to/from the target IP (78.27.71.99)

• Save them to a .pcap file for further analysis

Result: failed due to Insufficient permissions

Packet Capture and Traffic Analysis with Wireshark

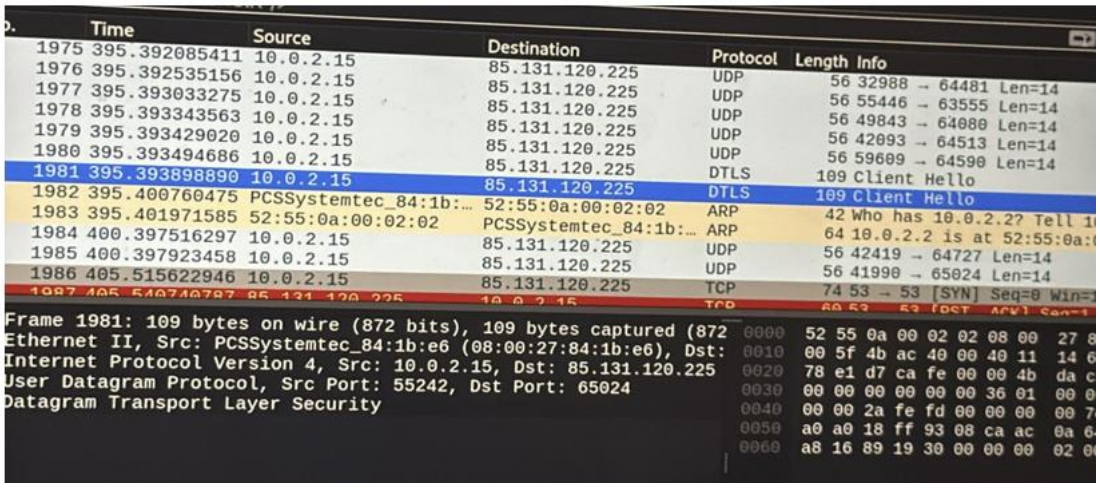


Figure 23. Wireshark Capture Showing DTLS Client Hello and Encrypted UDP Traffic (Device 2).

From figure 23, we observe DTLS traffic between

• Source: 10.0.2.15

• Destination: 85.131.120.225

• Protocol: UDP/DTLS (Client Hello)

This indicates

• Encrypted session initiation

• UDP-based telemetry typical of IoT behavior

4.3 Ethical and Legal Considerations

All tests were conducted on devices owned and controlled by the researchers. No public systems were accessed. The experiments adhered to ethical hacking standards from OWASP and CEH, and followed legal best practices including respect for data privacy and system integrity.

.

5 DISCUSSION

This chapter presents and discusses the key findings from the penetration tests performed on the target IoT devices. The objective of this study was to simulate potential attack vectors using ethical hacking methods and in so doing, identify the security weaknesses (or strengths) in each device, and interpret their implications in terms of smart home security. All tests were performed in a controlled environment using Kali Linux.

This research has shown that some smart home IoT devices can be susceptible to a range of security vulnerabilities. Through a structured penetration testing methodology using Kali Linux, it was demonstrated that these devices commonly expose insecure services, rely on outdated software, and might lack proper access control. Methods such as port scanning, service enumeration, OS fingerprinting, brute-force testing, and passive packet analysis revealed some security weaknesses that bad actors could exploit.

5.1 Findings from Different Testing Activities

The findings from the various tests carried out are briefly outlined below. The discussion here is structured according to the phases of the methodology.

5.1.1 Reconnaissance and Service Exposure

• Both devices exposed multiple open ports, including web interfaces (HTTP/HTTPS), DNS, and UPnP

• Device 1's use of outdated services (e.g., Intel UPnP SDK 1.2) and exposure of an HTTP admin interface indicates a high-risk posture

• Device 2 also exposed a PPTP VPN service and DNS fingerprint (DNA), which may reveal internal configurations

5.1.2 OS and Networking Stack Fingerprinting

• Both devices ran on outdated Linux kernels (2.4.x), which lack modern security features

• One-hop traceroute confirms physical or virtual proximity in the network, potentially simplifying attacker reach

5.1.3 Brute-force Vulnerability

• Hydra tests on the HTTP interface using brute-force attacks were inconclusive. While this doesn't reveal the strength or otherwise of the authentication measures of the device, it is recommended that proper authentication measures be enforced to prevent failures in future.

• The use of default or weak credentials remains a serious threat in consumer-grade IoT

5.1.4 Exploitation Observations

• Metasploit module listings show available RCE and UPnP exploits targeting similar devices

• Although no exploits were executed, the discovery process validates the risks of running exposed, outdated services

5.1.5 Packet-Level Traffic Observations

• Wireshark capture showed encrypted DTLS traffic, confirming secure session attempts

• UDP-based telemetry is observable, but payload content remains protected

Passive traffic monitoring provides situational awareness but is constrained without elevated access.

5.1.6 Overall Security Posture and Recommendations

TABLE 5. Security State of Devices and Recommendations.

Observation	**Risk Level**	**Recommendation**
Open admin interface (HTTP	Medium	Enforce HTTPS, use authentication and access control
Outdated UPnP and Linux kernel	High	Upgrade firmware, disable unnecessary services
Brute-force test inconclusive	Medium	Enforce strong password policies, implement MFA
Service fingerprinting (DNS, HTTP headers)	Medium	Mask banner info, restrict external access
UDP traffic exposed	Low	Encrypt sensitive telemetry and use secure session protocols

5.2 Contributions to Study

This study contributes to the growing body of work on smart home cybersecurity by:

- Providing a replicable methodology for ethical penetration testing of IoT devices
- Demonstrating how readily available tools like Nmap, Hydra, and Metasploit can uncover real-world vulnerabilities
- Offering insight into the common misconfigurations and outdated protocols present in IoT environments

5.3 Limitations of the Study

The study was limited to two Smart Home devices within a controlled testbed, which may not capture the full range of complexities, diversities and configurations found in real-world smart homes. Additionally, the need for elevated privileges for some tools, such as tcpdump, hindered complete traffic analysis. These limitations highlight the need for broader testing in future studies.

5.4 Recommendations

Based on the analysis, the following recommendations are offered:

• **For Manufacturers**: Ensure devices are equipped with strong default configurations, updated firmware, and built-in secure communication protocols like HTTPS and DTLS before shipment

• **For End Users**: Change default passwords, disable unused services, and apply firmware updates regularly

• **For Researchers**: Extend testing to more device categories (e.g., smart locks, smart plugs), and explore automation frameworks for vulnerability detection

• **For Policy Makers**: Encourage or enforce minimum security baselines for consumer-grade IoT devices through certification or labelling systems

• Reinforcing the importance of passive and active reconnaissance as part of a holistic security assessment.

5.5 Final Thoughts

This project has emphasized the real and present risks associated with unsecured smart home devices. As adoption of IoT systems grows, it is critical to prioritize cybersecurity at every stage—from design and deployment to use and maintenance. Also, ethical penetration testing remains a key tool for identifying vulnerabilities and helping stakeholders build safer, more resilient IoT ecosystems.

REFERENCES

Das, L., Anand, P., Anjum, A., Aarif, M., Maurya, N., & Rana, A. (2023, December). The impact of smart homes on energy efficiency and sustainability. In *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)* (Vol. 10, pp. 215-220). IEEE.

Ahmed, S., & Khan, M. (2023). Securing the Internet of Things (IoT): A comprehensive study on the intersection of cybersecurity, privacy, and connectivity in the IoT ecosystem. *AI, IoT and the Fourth Industrial Revolution Review*, *13*(9), 1-17. https://scicadence.com/index.php/AI-IoT-REVIEW/article/view/13. Accessed 16.01.2024.

Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, *54*(15), 2787-2805.

Adnyana, I.G., Thalib, E.F., Harum, M.A., Nagas, M.A.C. and Jawa, M.W., 2023. A Discussion of Malware Attacks Targeting Smart Homes and Connected Devices: Investigating Cybersecurity Risks in Everyday Living. *Journal of Digital Law and Policy*, *3*(1), pp.13-25.

Javed, M. H. (2023). Internet of Things Hacking: Ethical Hacking of a Smart Camera.

Reddy, Y.A., Kumar, C.A., Rukmani, P. and Ganapathy, S., 2022. A new compromising security framework for automated smart homes using vapt. In *Handbook of Research of Internet of Things and Cyber-Physical Systems* (pp. 337-366). Apple Academic Press.

Shafiq, M., Gu, Z., Cheikhrouhou, O., Alhakami, W., & Hamam, H. (2022). The Rise of “Internet of Things”: Review and Open Research Issues Related to Detection and Prevention of IoT-Based Security Attacks. *Wireless Communications and Mobile Computing*, *2022*(1), 8669348.

De Neira, A.B., Kantarci, B. and Nogueira, M., 2023. Distributed denial of service attack prediction: Challenges, open issues and opportunities. *Computer Networks*, *222*, p.109553.

Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., ... & Zhou, Y. (2017). Understanding the mirai botnet. In *26th USENIX security symposium (USENIX Security 17)* (pp. 1093-1110).

Bygrave, L. A. (2022). Security by design: Aspirations and realities in a regulatory context. *Oslo Law Review*, (3), 126-177.

Thomas, G., Burmeister, O., & Low, G. (2019). The Importance of Ethical Conduct by Penetration Testers in the Age of Breach Disclosure Laws. *Australasian Journal of Information Systems*, *23.*

Zaidan, A.A., Zaidan, B.B., Qahtan, M.Y., Albahri, O.S., Albahri, A.S., Alaa, M., Jumaah, F.M., Talal, M., Tan, K.L., Shir, W.L. and Lim, C.K., 2018. A survey on communication components for IoT-based technologies in smart homes. *Telecommunication Systems*, *69*, pp.1-25.

Papatsimouli, M., Lazaridis, L., Ziouzios, D., Dasygenis, M., & Fragulis, G. (2022). Internet of things (IOT) awareness in Greece. In *SHS Web of Conferences* (Vol. 139, p. 03013). EDP Sciences.

Magara, T., & Zhou, Y. (2024). Internet of things (IoT) of smart homes: privacy and security. *Journal of Electrical and Computer Engineering*, *2024*(1), 7716956.

Hewitt, M., & Cunningham, H. (2022). Taxonomic classification of IoT smart home voice control. *arXiv preprint arXiv:2210.15656*

Conti, M., Dragoni, N., & Lesyk, V. (2016). A survey of man in the middle attacks. *IEEE communications surveys & tutorials*, *18*(3), 2027-2051.

McKay, K., & Cooper, D. (2017). *Guidelines for the selection, configuration, and use of transport layer security (TLS) implementations* (No. NIST Special

Publication (SP) 800-52 Rev. 2 (Draft)). National Institute of Standards and Technology.

Anand, N., Parwekar, P., & Sharma, A. (2024). Optimized LoRaWAN Architectures: Enhancing Energy Efficiency and Long-Range Connectivity in IoT Networks for Sustainable, Low-Power Solutions and Future Integrations with Edge Computing and 5G. *Journal of Intelligent Systems & Internet of Things*, *13*(2).

Dechand, S., Naiakshina, A., Danilova, A., & Smith, M. (2019, June). In encryption we don't trust: The effect of end-to-end encryption to the masses on user perception. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)* (pp. 401-415). IEEE.

Barua, A., Al Alamin, M. A., Hossain, M. S., & Hossain, E. (2022). Security and privacy threats for bluetooth low energy in iot and wearable devices: A comprehensive survey. *IEEE Open Journal of the Communications Society*, *3*, 251-281.

Dragomir, D., Gheorghe, L., Costea, S., & Radovici, A. (2016, September). A survey on secure communication protocols for IoT systems. In *2016 international workshop on Secure Internet of Things (SIoT)* (pp. 47-62). IEEE.

Asonze, C.U., Ogungbemi, O.S., Ezeugwa, F.A., Olisa, A.O., Akinola, O.I. and Olaniyi, O.O., 2024. Evaluating the trade-offs between wireless security and performance in IoT networks: A case study of web applications in AI-driven home appliances. *Available at SSRN 4927991.*

Maleh, Y., & Ezzati, A. (2016). Towards an efficient datagram transport layer security for constrained applications in internet of things. *International Review on Computers and Software*, *11*(7), 611-621.

Al-Masri, E., Kalyanam, K. R., Batts, J., Kim, J., Singh, S., Vo, T., & Yan, C. (2020). Investigating messaging protocols for the Internet of Things (IoT). *IEEE Access*, *8*, 94880-94911.

Adiwal, S., Ahmed, S. S., Rajendran, B., Misbahuddin, M., & Sudarsan, S. D. (2024, September). Role of PKI in Securing AMQP Communication. In *2024 IEEE International Conference on Public Key Infrastructure and its Applications (PKIA)* (pp. 1-8). IEEE.

Hintaw, A. J., Manickam, S., Karuppayah, S., Aladaileh, M. A., Aboalmaaly, M. F., & Laghari, S. U. A. (2023). A robust security scheme based on enhanced symmetric algorithm for MQTT in the Internet of Things. *IEEE Access*, *11*, 43019-43040.

Iivari, A., Väisänen, T., Ben Alaya, M., Riipinen, T., & Monteil, T. (2014). Harnessing xmpp for machine-to-machine communications & pervasive applications. *Journal of Communications Software and Systems*, *10*(3), 163-178.

Mansfield-Devine, S. (2021). Who's that knocking at the door? The problem of credential abuse. *Network Security*, *2021*(2), 6-15.

Luna, S. I. (2020). *Man-in-the–Middle Attack* (Doctoral dissertation, East West University)

Malhotra, P., Singh, Y., Anand, P., Bangotra, D. K., Singh, P. K., & Hong, W. C. (2021). Internet of things: Evolution, concerns and security challenges. *Sensors*, *21*(5), 1809.

Abed, A. K., & Anupam, A. (2023). Review of security issues in Internet of Things and artificial intelligence-driven solutions. *Security and Privacy*, *6*(3), e285.

Aly, M., Khomh, F., Haoues, M., Quintero, A., & Yacout, S. (2019). Enforcing security in Internet of Things frameworks: A systematic literature review. *Internet of Things*, *6*, 100050.

Sarker, K.U., Yunus, F. and Deraman, A., 2023. Penetration taxonomy: A systematic review on the penetration process, framework, standards, tools, and scoring methods. *Sustainability*, *15*(13), p.10471.

Weir, C., Becker, I. and Blair, L., 2023. Incorporating software security: using developer workshops to engage product managers. *Empirical Software Engineering*, *28*(2), p.21.

Taiwo, O., Ezugwu, A. E., Oyelade, O. N., & Almutairi, M. S. (2022). Enhanced intelligent smart home control and security system based on deep learning model. *Wireless communications and mobile computing*, *2022*(1), 9307961.

Shafik, W., Matinkhah, S.M. and Shokoor, F., 2023. Cybersecurity in unmanned aerial vehicles: A review. *International Journal on Smart Sensing and Intelligent Systems*, *16*(1)

Soepeno, R.A.A.P., 2023. Wireshark: An Effective Tool for Network Analysis. *CYBV-Introd. Methods Netw. Anal*, pp.1-15.

Mohammed, A. H. Y., Dziyauddin, R. A., & Latiff, L. A. (2023). Current multi-factor of authentication: Approaches, requirements, attacks and challenges. *International Journal of Advanced Computer Science and Applications*, *14*(1)

Tu, Y., Rampazzi, S., Hao, B., Rodriguez, A., Fu, K., & Hei, X. (2019, November). Trick or heat? Manipulating critical temperature-based control systems using rectification attacks. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (pp. 2301-2315).

Van Woudenberg, J., & O'Flynn, C. (2021). *The hardware hacking handbook: breaking embedded security with hardware attacks*. No Starch Press.

Zandberg, K., Schleiser, K., Acosta, F., Tschofenig, H., & Baccelli, E. (2019). Secure firmware updates for constrained iot devices using open standards: A reality check. *IEEE access*, *7*, 71907-71920.

Xenofontos, C., Zografopoulos, I., Konstantinou, C., Jolfaei, A., Khan, M. K., & Choo, K. K. R. (2021). Consumer, commercial, and industrial iot (in) security: Attack taxonomy and case studies. *IEEE Internet of Things Journal*, *9*(1), 199-221.

Sadhu, P. K., Yanambaka, V. P., & Abdelgawad, A. (2022). Internet of things: Security and solutions survey. *Sensors*, *22*(19), 7433.

Chaudhary, R., & Kumar, N. (2019). LOADS: Load optimization and anomaly detection scheme for software-defined networks. *IEEE Transactions on Vehicular Technology*, *68*(12), 12329-12344.

Wilhelm, T. (2025). *Professional penetration testing: Creating and learning in a hacking lab*. Elsevier.

Ding, A. Y., De Jesus, G. L., & Janssen, M. (2019, September). Ethical hacking for boosting IoT vulnerability management: A first look into bug bounty programs and responsible disclosure. In *Proceedings of the Eighth International Conference on Telecommunications and Remote Sensing* (pp. 49-55).

Qusef, A., & Alkilani, H. (2022). The effect of ISO/IEC 27001 standard over open-source intelligence. *PeerJ Computer Science*, *8*, e810.

Siboni, S., Sachidananda, V., Meidan, Y., Bohadana, M., Mathov, Y., Bhairav, S., ... & Elovici, Y. (2018). Security testbed for Internet-of-Things devices. *IEEE transactions on reliability*, *68*(1), 23-44.

ChatGPT, 2025. Common Cybersecurity Threats in Smart Homes [AI-generated image]. Created with OpenAI ChatGPT on 22 May 2025.

OpenAI (2025) IoT's Arrival in Smart Homes [AI-generated image]. Created using DALL·E. URL : https://openai.com/dall-e Created: 22.5.2025.

ENISA (European Union Agency for Cybersecurity) (2021) IoT Security Baseline. URL: https://www.enisa.europa.eu/publications/iot-security-baseline Accessed: 25.5.2025

Cloudflare. (n.d.). What is HTTPS? | How HTTPS works. Cloudflare Learning Center. URL: https://www.cloudflare.com/learning/ssl/what-is-https/ Accessed: 22.5.2025

The Things Network. (n.d.). LoRaWAN Security. The Things Network Documentation. URL: https://www.thethingsnetwork.org/docs/lorawan/security/ Accessed: 22.5.2025

Zigbee Alliance. (2024). Zigbee Protocol Specification and Security Guidelines. URL: https://csa-iot.org Accessed: 22.5.2025

MQTT.org. (n.d.). MQTT Security Fundamentals. URL: https://mqtt.org/mqtt-security-fundamentals/ Accessed: 22.5.2025

Open Web Application Security Project (OWASP). (n.d.). Bluetooth Security. URL: https://owasp.org/www-community/attacks/BlueTooth_attacks. Accessed: 22.5.2025

CyberMaxx (2023) What is Penetration Testing? URL: https://www.cybermaxx.com/resources/what-is-penetration-testing/ Accessed: 22.5.2025

National Institute of Standards and Technology (NIST), 2023. An Introduction to Information Security (SP 800-12 Rev. 2). [online] U.S. Department of Commerce. URL: https://doi.org/10.6028/NIST.SP.800-12r2 Accessed: 22.5.2025

APPENDICES

Appendix 1 List of Figures

Appendix 2 List of Tables