Linnéuniversitetet
Kalmar Växjö

Bachelor Thesis in Computer Science

Case Study: Assessing the Security of a ZigBee Smart Home Network

Author: Robal Saker and Obaida Abu Issa
Supervisor: Ola Flygt
Semester: VT23
Discipline: Computer Science

Abstract

Utilizing the ZigBee protocol is pervasive in the context of smart homes, offering substantial convenience to individuals. However, smart home devices commonly handle significant quantities of real-world information, potentially giving rise to concerns related to information leakage. Therefore, in this study, we assess the security of a ZigBee smart home network by identifying potential vulnerabilities and conducting a penetration test on the network. In addition, this study compared the potential damage inflicted on the ZigBee network by technical and non-technical users. Identifying the potential vulnerabilities was carried out by following a systematic literature review approach, while the penetration testing method was conducted with the help of a case study. The smart home network consisted of a gateway, a smart light bulb, a temperature and humidity sensor, and a motion sensor. The results show a vulnerability in the ZigBee protocol where the technical user could compromise all the security keys. However, the non-technical user was not able to compromise the ZigBee network. Consequently, the security of ZigBee-based smart devices still needs to be further investigated and strengthened. Finally, we discussed the future of the ZigBee network and the optimal scenarios for deploying it.

Keywords

ZigBee, vulnerabilities, smart home, technical user, non-technical user

Contents

1 Introduction **4**
- 1.1 Background . . . 4
- 1.2 Related Work . . . 4
- 1.3 Problem Formulation . . . 5
- 1.4 Motivation . . . 5
- 1.5 Results . . . 5
- 1.6 Limitation . . . 6
- 1.7 Target Group . . . 6
- 1.8 Outline . . . 6

2 Method **7**
- 2.1 Systematic Literature Review . . . 7
  - 2.1.1 Generating a Search Strategy . . . 7
  - 2.1.2 Study Selection Process . . . 8
  - 2.1.3 Data Extraction and Synthesis . . . 9
- 2.2 Case Study . . . 9
  - 2.2.1 Case Study Design and Preparation for Data Collection . . . 10
  - 2.2.2 Hardware and Software Selection Process for ZigBee Exploitation 10
  - 2.2.3 Methods Element . . . 10
  - 2.2.4 Collecting Evidence and Analysis . . . 13
- 2.3 Reliability and Validity . . . 13
- 2.4 Ethical Consideration . . . 14

3 Technical Background for ZigBee Smart Home Environment **15**
- 3.1 Smart Home . . . 15
- 3.2 IEEE 802.15.4 Standard . . . 15
  - 3.2.1 PHY . . . 15
  - 3.2.2 MAC Sublayer . . . 16
  - 3.2.3 Security Services . . . 16
- 3.3 ZigBee Protocol . . . 17
  - 3.3.1 ZigBee Architecture Stack . . . 18
    - 3.3.1.1 Application (APL) Layer . . . 18
    - 3.3.1.2 Network Layer . . . 21
    - 3.3.1.3 Security Services . . . 23

4 Results **28**
- 4.1 Systematic Literature Review Process . . . 28
- 4.2 ZigBee Vulnerabilities . . . 29
- 4.3 Non-Technical User Penetration Case . . . 30
- 4.4 Technical User Penetration Case . . . 30

5 Analysis **35**
- 5.1 Systematic Literature Review Process . . . 35
- 5.2 Research Question 1 . . . 35
- 5.3 Research Question 2 . . . 36
  - 5.3.1 Non-Technical User Penetration Case . . . 36
  - 5.3.2 Technical User Penetration Case . . . 37

6 Discussion **39**
6.1 Research Question 3 39
6.2 Related Work and Findings 39

7 Conclusion and Future Work **40**
7.1 Conclusion 40
7.2 Future Work 40

Appendices **46**
A Systematic Literature Review Results 46
B Hardware and Software Tools 46
C Hash Values 46
D Installation of AT 47
E Wireshark Traffic 49

Definitions of Terms

The defined terms in this section are used to construct validity by mitigating unintended interpretations of the terms. These terms have been taken from the ZigBee specification [1].

Application profile: "This is a collection of device descriptions, which together form a cooperative application. For instance, a thermostat on one node communicates with a furnace on another node. Together, they cooperatively form a heating application profile."

Application object: "This is a component of the top portion of the application layer defined by the manufacturer that actually implements the application."

Attribute: "This is a data entity which represents a physical quantity or state. This data is communicated to other devices using commands."

Binding: "This is the creation of a unidirectional logical link between a source endpoint/cluster identifier pair and a destination endpoint, which may exist on one or more devices."

Broadcast: "This is the transmission of a message to every device in a particular PAN belonging to one of a small number of statically defined broadcast groups, for example all routers, and within a given transmission radius measured in hops."

Cluster: "This is an application message, which may be a container for one or more attributes. As an example, the ZigBee Device Profile defines commands and responses. These are contained in Clusters with the cluster identifiers enumerated for each command and response. Each ZigBee Device Profile message is then defined as a cluster. Alternatively, an application profile may create sub-types within the cluster known as attributes. In this case, the cluster is a collection of attributes specified to accompany a specific cluster identifier (sub-type messages.)"

Cluster identifier: "This is a reference to an enumeration of clusters within a specific application profile or collection of application profiles. The cluster identifier is a 16-bit number unique within the scope of each application profile and identifies a specific cluster. Conventions may be established across application profiles for common definitions of cluster identifiers whereby each application profile defines a set of cluster identifiers identically. Cluster identifiers are designated as inputs or outputs in the simple descriptor for use in creating a binding table."

End application: "This is for applications that reside on endpoints 1 through 254 on a Device. The end applications implement features that are non-networking and ZigBee protocol related. Endpoints 241 through 254 shall only be used by the End application with approval from the ZigBee Alliance. The Green Power cluster, if implemented, SHALL use endpoint 242."

Endpoint: "This is a particular component within a unit. Each ZigBee device may support up to 254 such components."

Endpoint address: "This is the address assigned to an endpoint. This address is assigned in addition to the unique, 64-bit IEEE address and 16-bit network address."

Information base: "This is a collection of variables that define certain behavior in a layer. These variables can be specified or obtained from a layer through its management service."

MCPS-SAP: "Medium access control common part sub-layer service access point."

Network address: "This is the address assigned to a device by the network layer and used by the network layer for routing messages between devices."

Node: "This is a collection of independent device descriptions and applications residing in a single unit and sharing a common 802.15.4 radio."

PAN coordinator: "the principal controller of an IEEE 802.15.4-based network that

is responsible for network formation. The PAN coordinator must be a full function device (FFD)."

Protocol data unit: "the unit of data that is exchanged between two peer entities."

Radio: "the IEEE 802.15.4 radio that is part of every ZigBee device."

Service discovery: "the ability of a device to locate services of interest."

ZigBee device object: "the portion of the application layer responsible for defining the role of the device within the network (for example, ZigBee coordinator or end device), initiating and/or responding to binding and discovery requests, and establishing a secure relationship between network devices."

List of Abbreviations

AED	Advanced Encryption Algorithm
AIB	APS Information Base
AODVjr	Ad hoc On-Demand Distance Vector junior
APL	Application
APS	Application Support Sub-layer
APSDE	APS Data Entity
APSDE-SAP	APS Data Entity SAP
APSME	APS Management Entity
APSME-SAP	APS Management Entity SAP
APDU	Application Level PDU
AT	Automated Tools
CBC-MAC	Cipher Block Chaining Message Authentication Code
CCA	Clear Channel Assessment
CS	Computer Science
DOS	Denial of Service
ED	Energy Detection
FFD	Full-Function Device
GTS	Guaranteed Time Slot
ID	Identifier
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
LQI	Link Quality Indication
LR-WPAN	Low-Rate Wireless Personal Area Network
MAC	Medium Access Control
MACs	Medium Authentication Codes
MIC	Message Integrity Check
MLME-SAP	MAC Sublayer Management Entity SAP
MitM	Man-in-the-Middle
MPDUs	MAC Protocol Data Units
MQTT	Message Queuing Telemetry Transport
NIB	Network Information Base
NLDE	NWK Layer Data Entity
NLME	NWK Layer Management Entity
NSLA	Nedis Smart Life Application
NWK	Network
OS	Operating System

PAN	Personal Area Network
PDU	Protocol Data Unit
PHY	Physical
PSB	Philips Smart Bulb
RF	Radio Frequency
RFD	Reduced-Function Device
RQ	Research Question
SAP	Service Access Point
SSP	Security Service Provider
TaH	Temperature and Humidity
ZAF	ZigBee Application Framework
ZCL	ZigBee Cluster Library
ZDO	ZigBee Device Object
ZDP	ZigBee Device Profile

1 Introduction

Technology has made significant progress and continues to advance, aiming to enhance various aspects of society. Experts define the Internet of Things (IoT) as "the network of physical objects—"things"—that are embedded with sensors, software, and other technologies to connect and exchange data with other devices and systems over the internet" [2]. IoT devices range from simple consumer products like smartwatches or light bulbs to complex industrial equipment like manufacturing machines, utility grids, agricultural drones, or glucose monitoring devices [3]. IoT emerged decades ago by developing the first internet-connected toaster in 1990 [4]. However, the 2000s saw widespread adoption of IoT devices, driven by advancements in wireless technology and sensor miniaturization [3]. Consumer IoT devices have recently seen significant development due to the growing demand for smart living solutions. According to Xiaomi's 2021 annual statistics report, experts expect IoT devices to surpass 75 billion by 2025 [5]. IoT technology offers improved efficiency and productivity, enhanced automation, and greater access to real-time data [3]. However, concerns about IoT security, privacy, and data ownership persist [3]. ZigBee, commonly used in smart home solutions, offers low-power and low-cost wireless communication [6]. ZigBee networks, known for their low complexity and stability, provide a low-security network that attackers can easily exploit. Consequently, this 15 HEC bachelor thesis in Computer Science (CS) aims to assess the security of a ZigBee smart home network against technical and non-technical intruders.

1.1 Background

The ZigBee Alliance developed the ZigBee protocol, creating an economical and energy-efficient wireless communication standard capable of bidirectional transmission. Various sectors will integrate ZigBee standard-based implementations, including consumer electronics, home and building automation, industrial controls, PC peripherals, medical sensor applications, toys, and games. The ZigBee network comprises three devices: coordinator, router, and end device. The coordinator establishes, implements, and manages the network [6]. The router is an intermediate node to transmit data packets [6]. The end device, typically a sensor device, monitors and collects environmental data [6]. ZigBee networks support three topologies: star, mesh, and cluster tree networks, as seen in Figure 3.3 [7]. Following the IEEE 802.15.4 standard for low-rate wireless transmission [8], ZigBee networks play a crucial role in smart home systems, providing robust and efficient communication for a wide array of devices [3, 9]. The mesh topology, predominantly used in smart homes, offers self-healing and extended coverage capabilities, ensuring network robustness even when individual devices fail [10].

1.2 Related Work

ZigBee is an extensively researched subject, with ongoing efforts to review its historical, current, and potential future developments. The related work section of this paper compiles the most relevant contributions to this study. In 2019, Khanji et al. published research evaluating ZigBee's security performance in smart applications like automatic metering and industrial automation [10]. The study highlights the need for robust security enhancements due to ZigBee's vulnerability to network attacks. The paper provides an overview of ZigBee's security architecture, discusses its vulnerabilities, and suggests literature-based countermeasures and controls. It also points out ZigBee's persistent susceptibility to network attacks due to its low computing power. The paper concludes by emphasizing the importance of continued research in this field. In 2023, Praveen

et al. examined the security challenges in the Internet of Things (IoT) arena, particularly assessing the effectiveness of open-source tools for testing IoT networks using ZigBee and MQTT protocols [11]. The study highlights the evolution of IoT attack tools and the need for improved automated systems for attack detection and prevention. It also points out the need for updated and maintained open-source tools for testing IoT security and proposes features for developing new and enhancing existing tools. In 2015, Bin Karnain and Bin Zakaria proposed various methods to enhance ZigBee network security in Smart Home environments [12]. One approach introduces a new security scheme involving XOR calculations and time-based key updates to protect against active and passive attacks. Another method suggests using ZigBee's Received Signal Strength to defend against spoofing attacks by monitoring network nodes' signal strength. Additionally, a proposal aims to strengthen the Ad hoc On-Demand Distance Vector junior (AODVjr) routing protocol in ZigBee networks through a novel key distribution scheme that integrates the Diffie-Hellman algorithm. However, these methods still need to address forward security fully, which is essential for securing communications when devices leave the network.

1.3 Problem Formulation

This study aims to evaluate the security of the ZigBee network and its penetrability. Recent literature on ZigBee networks reveals specific vulnerabilities in particular protocols of the ZigBee stack. We aim to study the ZigBee protocol stack to determine if it is easily exploitable, even by someone without knowledge about ZigBee or cybersecurity. From the points mentioned above, this study aims to answer the following research questions:

- RQ1: What are the vulnerabilities in the ZigBee protocol, and how frequently do they occur?
- RQ2: What are the results of conducting penetration testing on a ZigBee smart home network for technical and non-technical intruders?
- RQ3: Where is the future of ZigBee network heading and the best place to deploy them with its current vulnerabilities?

1.4 Motivation

The current state of ZigBee security is being heavily researched and constantly improving. Recent studies conclude that ZigBee networks still have weaknesses or lack security that must be addressed and enhanced [6, 13]. This study aims to build upon the previous research to review ZigBee networks' security for future researchers by assessing current vulnerabilities to highlight the state of the art of ZigBee. Thus, new knowledge is necessary for industrial, commercial, personal, and critical infrastructures using ZigBee networks and IoT devices to operate more securely. The new knowledge will lead to a potentially safer society and evolution in the world's technological sector.

1.5 Results

This study employs an empirical approach to evaluate the security of a ZigBee smart-home solution composed of a coordinator and multiple end devices based on the IEEE 802.15.4 standard [8]. The study uses a two-phase experimental case study approach and a systematic literature review encompassing studies from 2019 to 2023 to evaluate the network's attack vulnerabilities. In the first phase, a non-technical user with no background in CS attempts to exploit the network using penetration tools obtained

from online sources. This phase aims to assess the ease with which a non-technical user can compromise the network and identify potential security loopholes. The second phase involves a technical user using information gathered from one or more applications, tools, and web resources to challenge the security of the network devices. This phase aims to evaluate the effectiveness of the security measures employed by the network and identify any weaknesses that a technical intruder could exploit. The study seeks to provide insights into the security risks associated with ZigBee smart-home solutions and identify possible countermeasures to mitigate these risks. The findings of this study can inform the development of secure smart-home solutions and contribute to the ongoing discussion on the security of IoT devices. By evaluating the security of a ZigBee smart-home solution, this study aims to raise awareness about the importance of security in IoT devices and encourage adopting best practices in developing smart-home solutions.

1.6 Limitation

This study involves managing three primary limitations. The first limitation pertains to insufficient time to conduct a complete penetration test on all areas of the ZigBee network or collect sufficient test subjects acting as technical and non-technical intruders. Therefore, there will be a focus on exploiting specific vulnerabilities to yield sufficient results. The lack of test subjects is explained in chapter 2.2.3. The second limitation involves the inability to obtain specific IoT devices necessary for the research due to funding limitations or unavailability. To mitigate this limitation, we will determine the availability of all devices within the budget range and construct the network around those devices. Finally, the last limitation is the lack of understanding of ZigBee network architecture or insufficient knowledge of best practices for designing secure smart home networks, which could lead to developing an architecture that is vulnerable to attack. Thus, proper research on ZigBee networks will be conducted to build an architecture with the best possible practices.

1.7 Target Group

The target groups of this study are individuals and organizations within the CS community interested in the security of IoT devices and networks. These groups include researchers, academics, students, developers, and practitioners in computer security, network security, and IoT. It also encompasses government agencies, non-profit organizations, and the general public interested in IoT devices and networks' security and privacy implications. The study's findings and recommendations have the potential to benefit a wide range of stakeholders, including those in academia, industry, and government.

1.8 Outline

The structure of this paper is delineated as follows: Chapter 2 explains the research methodologies used in this study, including its execution, and examines the aspects of reliability, validity, and ethical considerations. Chapter 3 establishes the theoretical framework, focusing on specific topics such as smart home technology, IEEE 802.15.4 standards, and the ZigBee protocol. Chapter 4 presents the empirical findings of the study. Subsequently, Chapter 5 dives into the analysis of these results. Chapter 6 discusses the findings, contrasting them with outcomes from related works, and evaluates the generalizability and validity of these findings. The paper concludes with Chapter 7, which synthesizes the conclusions drawn from the results and contemplates potential areas for future research.

2 Method

This research utilizes two methodological frameworks: a systematic literature review and a case study. The structured analysis of previous literature will help resolve RQ1, while the case study will provide necessary insights for addressing RQ2. The discussion section will answer RQ3 based on the results of RQ1 and RQ2.

2.1 Systematic Literature Review

The primary goal of RQ1 is to provide foundational knowledge for future research on potential vulnerabilities in the smart home ZigBee network by prioritizing them based on impact. RQ3 will use the knowledge from RQ1 to discuss the future of ZigBee networks and the best scenarios for their deployment. RQ1 aims to provide background for future studies, which is one of the reasons for using the literature review method [14]. We chose the systematic literature review as the best method to answer RQ1. Broadly, a systematic literature review is a secondary investigation aimed at mapping, identifying, and appraising existing research related to a particular subject [14]. This part of the study adopts an iterative methodology, following the approaches shown by both Kitchenham and Charters and Xiao et al., wherein the investigation divides into three principal phases, each containing specific activities [14], [15]. Figure 2.1 depicts the process, starting with a planning phase and then the review conduction. These phases and activities are iterative, enabling the researcher to return to any previous phase or activity as needed. The final phase involves presenting the review and disseminating its findings.

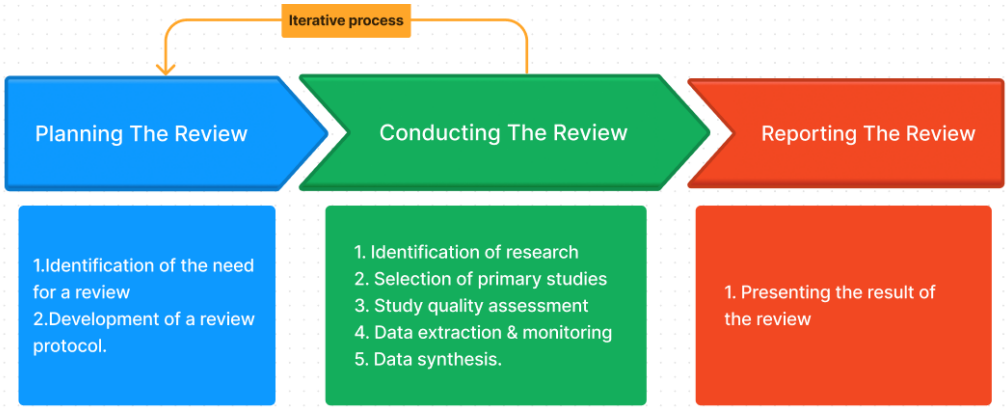

Figure 2.1: Summary of the systematic literature review process

The initial step in the planning phase involves identifying the need for a review. The introductory chapter of this study has already provided a comprehensive rationale and met the necessary prerequisites. Subsequently, developing a meticulous review protocol becomes pivotal, serving as a pre-established framework that outlines the methodologies for conducting the review. This protocol plays a critical role in reducing potential biases and enhancing the overall reliability of the study [14, 15]. The review protocol includes a detailed exposition that explains the implementation strategies for conducting and presenting the review. The subsequent sections of this chapter will elaborate on these strategies in greater detail.

2.1.1 Generating a Search Strategy

The search strategy is critical, guiding researchers on the appropriate locations and methodologies for conducting searches, focusing on transparency and applicability [14]. For this study, we searched using two electronic databases, namely IEEE Xplore and

ACM Digital Library, considering the specialized nature of our case study and time constraints. We selected these databases for their extensive technology and computer science coverage, ensuring access to high-quality, peer-reviewed content essential for our study's credibility. Our case study methodology's detailed focus required these databases to provide relevant and comprehensive material, enabling an efficient and in-depth review within our limited timeframe. Additionally, these databases played a crucial role in triangulating our case study findings, thereby enhancing the precision of our research. This focused approach, which prioritized quality over quantity, was crucial given our time constraints and aligned with our goal of maintaining thoroughness and integrity in our research. The selection of IEEE Xplore and ACM Digital Library thus supported a thorough, relevant, and efficient literature review, perfectly complementing our study's methodology and objectives. We identified suitable search words during initial exploratory searches to investigate the research topic. The main goal was to create search strings that effectively addressed this study's research questions. Simple search strings often led to favorable outcomes during the initial exploratory searches. Therefore, we combined words within the same concept using the OR operator while combining words from different concepts typically involving the AND operator. We conducted a series of test searches for each database to refine the search strings and identify the word combinations that produced the most favorable results. We subjected the final search strings, presented in Table 2.1, to a filtering process to include only studies published between 2019 and 2023.

Table 2.1: Shows the search string corresponding to each database used and how many searches were conducted.

Database	Search String
ACM 1	[Title: zigbee] AND [Full Text: vulnerabilities] AND [E-Publication Date: (01/01/2019 TO 31/01/2023)]
ACM 2	[Title: zigbee] AND [Full Text: vulnerabilities] AND [All: smart home] AND [E-Publication Date: (01/01/2019 TO 31/01/2023)]
IEEE 1	("Document Title":zigbee) AND ("Full Text Only":"vulnerability" OR "Full Text Only":"vulnerabilities")
IEEE 2	("Document Title":zigbee) AND ("All Metadata":"smart home") AND ("Full Text Only":"vulnerability")

2.1.2 Study Selection Process

The study used a three-stage selection process to curate relevant literature for this CS academic thesis. In the first stage, we executed search queries in predefined databases (discussed in Chapter 2.1.1) and scrutinized the titles of the resulting studies. We retained studies with titles relevant to the specific search purpose. The second stage involved reevaluating the abstracts to confirm their relevance and verify the inclusion criteria evident in the abstracts. Finally, in the last stage, we obtained full texts of the selected studies. We conducted a comprehensive review to ensure compliance with all inclusion criteria and to perform a quality assessment. We approved studies that successfully passed all three stages for inclusion in this research. Developing a review protocol required making critical decisions concerning the inclusion criteria and quality assessment requirements [14, 15]. We formulated detailed checklists for inclusion criteria and quality assessment to facilitate the review process before starting the actual review. The primary objective of the selection and exclusion criteria was to identify relevant literature for this study's purpose. We present the specific criteria chosen for this review in Table 2.2. A study must have met any or all of the selection criteria while strictly adhering to the exclusion criteria to be considered for inclusion in the review.

Kitchenham and Charters identified two primary purposes of quality assessment within reviews: supporting the study selection process and aiding in data analysis [14]. This study primarily used quality assessment as a supplementary tool during the study

Table 2.2: Shows the selection and exclusion criteria for the systematic literature review.

Inclusion Criteria
Selection criteria
Studies that outline general or new vulnerabilities in the ZigBee protocol.
Studies that evaluate the state of the art of the ZigBee network.
Exclusion criteria
Studies that are published before 2019.
Studies that have not undergone a peer review process.
Either the same study or an alternative version of the study has already been included.

selection process. Once a study met the inclusion criteria, we evaluated it against the quality checklist to gauge its credibility and relevance for this review. To handle cases where binary yes or no responses in the quality assessment could have been more straightforward and to avoid potentially excluding valuable studies, we integrated a notes section into the checklist. Whenever a study displayed quality attributes that did not fit neatly into strict yes or no categories, we recorded corresponding notes and included them in the data extraction form. This approach allowed for a comprehensive assessment of potential quality variations during the data extraction and analysis stages, enabling a more nuanced comparison if necessary. After selecting studies from initial database searches, we employed a snowballing technique as a supplementary method, guided by Wohlin's established guidelines, starting with the selected studies [16]. In this process, backward snowballing involved reviewing the references of each selected study, while forward snowballing involved identifying articles that cited these studies. We subjected studies identified via snowballing to the same rigorous selection criteria as the initial phase. Thus, this snowballing method was a repeated iteration of the study selection process.

2.1.3 Data Extraction and Synthesis

Determining a data extraction strategy is a crucial step in designing the review protocol for this study before starting the review process [14, 15]. We developed a meticulously crafted data extraction form to reduce the potential impact of bias and ensure uniform extraction of essential data relevant to the research questions. This form will assist in systematically collecting and organizing necessary information from the selected studies. After completing the data extraction process, we subjected the gathered data to meticulous analysis and comparison, drawing insights from diverse studies to compile comprehensive results.

Table 2.3: Summary of the data extraction form includes the Data Item, Description, and Motive.

Data Item	Description	Motive
Content of the study	Examining the purpose of the study and analyzing its quality	RQ1
Threats and Vulnerabilities	The threats and vulnerabilities mentioned in the study and their impact	RQ1
Hardware	The hardware used in the ZigBee network and to compromise the network	RQ1
Software	Application used or developed in the study	RQ1

2.2 Case Study

The main objective of RQ2 is to penetrate the ZigBee smart home network through both a technical and non-technical user to see if we can influence or change the state of the art in the ZigBee protocol. The objective of RQ2 aligns with the guidelines of a case study

in CS [17]. Therefore, we chose the case study method as the best method to answer RQ2. According to Runeson and Höst, a case study "is an empirical method to investigate contemporary phenomena in their context" [17]. This study follows an exploratory case study approach, aiming to delve into the current situation to understand what is happening. Through this method, we aim to uncover new insights and generate ideas and hypotheses for further research [17]. Conducting the case study involves five significant processes.

1. Case study design: objectives are defined, and the case study is planned.
2. Preparation for data collection: procedures and protocols for data collection are defined.
3. Collecting evidence: execution with data collection on the studied case.
4. Analysis of collected data
5. Reporting

2.2.1 Case Study Design and Preparation for Data Collection

The design process includes the objective, the case, the theory, research questions, methods, and selection strategy as its main elements [17]. Chapter 1 has already thoroughly answered all the elements except the methods element. Therefore, this section will address the methods concerns. The methods element aims to identify how to collect the data [17]. To explain the method element, we must define and elaborate on the hardware and software used in the study, the architecture of the smart home network, the penetration test method, and the study subjects. Consequently, this will also address the preparation for data collection by defining all the procedures and protocols for the collection phase.

2.2.2 Hardware and Software Selection Process for ZigBee Exploitation

Many hardware devices use the ZigBee radio frequency to sniff and inject ZigBee packets. However, we observed a notable need for more such hardware when we conducted this research in Sweden. Additionally, the available hardware is prohibitively expensive, posing a significant challenge for this study, which needs more financial sponsorship. This shortage and high-cost limit the accessibility of these devices for personal use, especially considering the financial constraints. In light of this study's focus on assessing the ability of both technical and non-technical users to penetrate the ZigBee protocol, we established inclusion criteria to assist in selecting appropriate hardware, as outlined in Table 5. This hardware limitation then imposes constraints on the software capabilities of the Automated Tools (AT) we used to exploit the ZigBee protocol. Each AT's effectiveness inherently depends on the range of hardware it supports, further complicating the research process. Appendix B lists all the hardware and software used in this study.

2.2.3 Methods Element

- **Technical and Non-Technical User**
  Technical User: Indicates an individual with knowledge in various computing aspects, including software applications, hardware configurations, networks, programming languages, and system structures. Technical users are often CS professionals, such as engineers, developers, IT specialists, and system administrators. They can deploy software solutions, troubleshoot complex issues, and make informed decisions about system integration and optimization. Thus,

Table 2.4: Inclusion criteria for hardware and software selection

Criterion	Description
Compatibility with ZigBee Protocol	The hardware must be fully compatible with the ZigBee radio frequency standard.
Cost-Effectiveness	The hardware cost should be within the limited budget of $100, emphasizing affordability.
Availability in Sweden	Hardware must be readily available for purchase within Sweden to ensure timely acquisition for the study.
Support for AT	The hardware should be compatible with the AT used in this study for ZigBee protocol exploitation.
Technical Specifications	Adequate technical capabilities such as range, signal strength, and stability to ensure effective testing and analysis.

a non-technical user can become technical by acquiring the expertise above. The technical user can use any means possible to penetrate the ZigBee smart home topology, except for the AT.

Non-Technical User: Indicates an individual who needs to gain the skills of a technical user. The time limitation of this study led to a lack of subjects acting as non-technical users. Therefore, we decided to simulate a non-technical user using the automated tools mentioned in Table B.1. We assume that the non-technical user lacks the knowledge and skills to penetrate the ZigBee network. However, the non-technical user can use the internet to search for methods to penetrate the ZigBee smart home network. We will assess the effectiveness of these selected tools in compromising the security of a ZigBee smart home network. This evaluation will help us understand the practicality of the information available to non-technical users and its potential impact on the network's security.

- **Smart Home Setup**
  At the start of the data collection phase, we set up the testing environment by implementing the ZigBee network architecture within a smart home setup context. We constructed this architecture using the hardware components listed in Table 4, forming a star topology configuration. The ZigBee gateway, depicted in Figure 2.2, lies at the core of this topology and facilitates data transmissions from various end devices like the motion sensor, Temperature, and Humidity (TaH) sensor, and smart light, to the Nedis Smart Life Application (NSLA). An access point (WIFI) facilitates the data's conveyance to the NSLA, enabling the application to supervise and manipulate the functionality of these end devices. We established access to the gateway via internet connectivity. Moreover, the gateway is pivotal in establishing and distributing security keys. We evaluated the functional efficacy of the ZigBee Network illustrated in Figure 2.2 by interfacing with the NSLA. Later, we accessed the gateway's user interface to validate the operational integrity of the smart light's on/off functionality, the motion sensor's detection capabilities, and the accuracy of TaH readings.

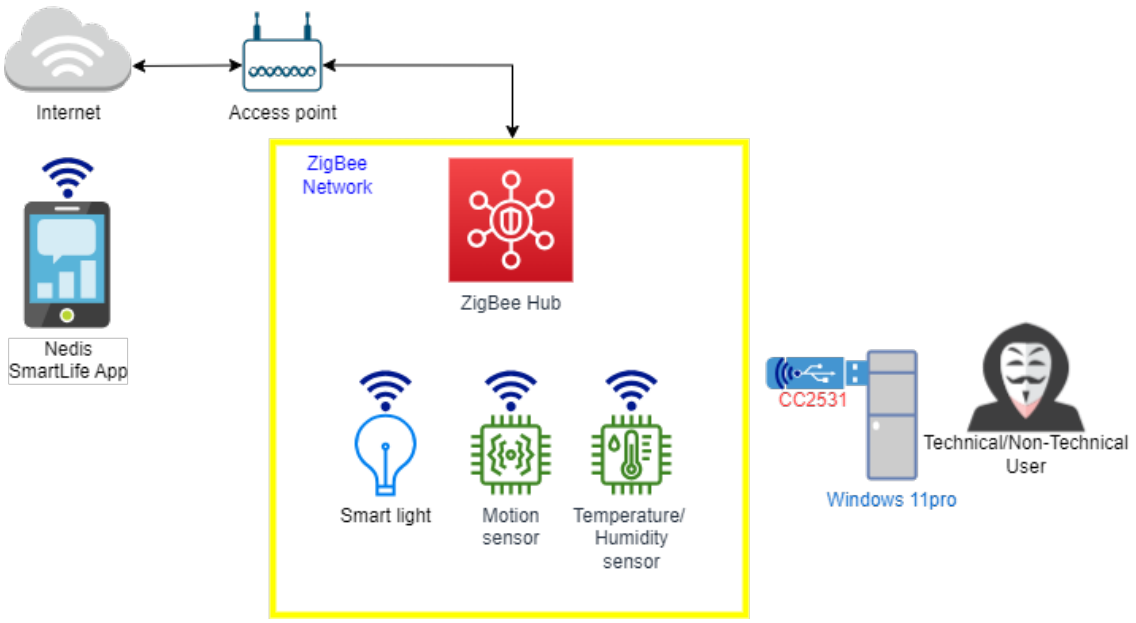

Figure 2.2: Attack scenario setup of the ZigBee smart home centralized topology.

- **Flashing the CC2531 USB Stick**
  The CC2531 will flashed and set up so it is ready to use by the technical and non-technical users. The steps for flashing the CC2531 USB stick are:

  1. Install the CC debugger driver on the PC and verify the installation by checking the device manager [18].
  2. Install SmartRF Flash Programmer version 1.
  3. Connect the CC debugger with the CC2531 downloader cable and CC2531 USB stick. Then, connect the USB to the PC.
  4. If the CC debugger light turns green, it signifies the successful establishment of a connection with the debugger.
  5. If the CC debugger light turns red, press the reset button on the CC debugger until the light turns green.
  6. Download the CC2531 DEFAULT 20211115 firmware.
  7. Run the SmartRF Flash Programmer and set up the configuration according to Figure 7.
  8. Press on "Perform actions", and the message "CC Debugger firmware update OK" will show up.

  Now, the CC2531 will be ready to sniff ZigBee packets after completing the above steps successfully.

- **ZBOSS and Wireshark**
  We will methodically install and configure ZBOSS and Wireshark to ensure their accurate operation, enabling technical and non-technical users to utilize them.

- **Ubuntu 23.10**
  We will also install and configure Ubuntu 23.10 to run Killerbee, BumbleBee, Z3SEC, and Zigbear.

2.2.4 Collecting Evidence and Analysis

This section discusses the data sources used for data collection. In this case study, we will apply a first-degree collection technique, allowing the researcher to directly contact the subjects and collect data in real-time [17]. The subjects are technical and non-technical users. We will place the users in the smart home environment to try collecting data by penetrating the ZigBee network. For the technical user, the ZBOSS sniffer will collect data and transmit it to Wireshark as data packets sniffed by the CC2531 during their penetration test. To ensure data reliability during collection, we will first initiate the ZBOSS sniffer and confirm the CC2531's interception of ZigBee packets by validating that the packets captured on both ZBOSS and Wireshark match identically before conducting any penetration tests. For the non-technical user, we will collect data using the AT. We will monitor the users directly throughout the data collection process to resolve any errors and ensure that the non-technical user only uses the AT. Therefore, we chose the first-degree technique for this scenario. We will save the collected data and generate a SHA-256 hash value to preserve its integrity (see Appendix C). We will duplicate the saved data for further analysis. After completing the evidence-collection stage, we will subject the gathered data to detailed qualitative analysis to draw relevant conclusions for reporting.

2.3 Reliability and Validity

Producing trustworthy results is crucial for research. Validity ensures that the conclusions are backed by the data and research method. Reliability focuses on the ability to replicate the study and obtain consistent results.

To mitigate reliability issues for the systematic literature review, a documented outline of the search strategy, selection process, and data extraction strategy is provided. Additionally, it includes comprehensive documentation of the search findings and all stages of the selection process outcomes to increase reliability.

To reduce reliability problems with the case study, a detailed documentation of the case study design, preparation for data collection, evidence collection, and analysis of collected data, following the guidelines of Runeson and Höst, is provided [17].

The study's validity focuses on three primary issues. The first concern, construct validity, relates to the interpretation of theoretical constructs within the paper. This report adopts well-established terms specific to ZigBee and cybersecurity to mitigate issues surrounding construct validity, primarily from existing research. However, unintended interpretations of specific terms or definitions might still exist, potentially affecting the report's validity.

The second issue, internal validity, depends on whether the collected data adequately supports the study's conclusions. We employed a predefined inclusion criteria checklist to mitigate bias risks in study selection. Furthermore, we used a quality assessment checklist to evaluate each included study, aiming to minimize biases introduced through the literature.

Within the case study, biases arise from multiple sources. One significant bias is the time constraint, resulting in limited subjects representing technical and non-technical users. To counteract this limitation, the researchers will act as representatives, potentially influencing the results and impacting the conclusions drawn. To mitigate this bias, we will confine non-technical user involvement to specified AT (see B.1). Researchers will closely monitor the case study's progress to ensure adherence to predetermined guidelines. However, inherent researcher involvement in monitoring leaves room for bias, a challenge tackled similarly to addressing reliability issues. Obaida Abu Issa will be the

technical user, while Robal Saker will be the non-technical user.

The last issue, external validity, concerns the extent to which the study's results can be generalized. To minimize external validity problems, we formulated the study's objectives and research questions within a feasible scope. The review process aimed to select various articles meticulously across different databases. However, resource limitations pose a risk of insufficient article inclusion, potentially affecting the generalizability of the results.

In empirical research, triangulation enhances precision. It involves approaching the studied subject from various angles, offering a comprehensive view. This approach is especially essential when relying predominantly on qualitative data, as in this study. Hence, we implemented methodological triangulation by combining a case study with a systematic literature review, enriching the research approach.

2.4 Ethical Consideration

Given the absence of interviews or surveys in this research endeavor, we did not collect sensitive data, thus negating considerations regarding confidentiality and participant consent. Nevertheless, a potential risk emerges concerning the selection process for studies in the literature review. The inclusion criteria required peer-reviewed studies, excluding findings from grey literature or preprints. Moreover, studies were confined to the period from 2019 to the present, using only two databases, consequently impacting the study's robustness and potentially overlooking significant contributions in the field.

In the case study's application, a noteworthy risk involves the detailed documentation of the penetration test. This comprehensive information could be misused, posing a threat if applied in real-life scenarios to compromise individuals' smart home networks. Therefore, we recommend using the insights from this case study exclusively within a personal smart home network or after obtaining explicit consent when engaging with any other smart home network.

3 Technical Background for ZigBee Smart Home Environment

This section offers the necessary technical background to understand the results and analysis presented later fully. It focuses on key concepts like smart homes, the IEEE 802.15.4 standard, and the ZigBee protocol.

3.1 Smart Home

The Oxford Dictionary defines a smart home as "a home equipped with lighting, heating, and electronic devices that can be controlled remotely by smartphone or computer: you can contact your smart home on the Internet to make sure the dinner is cooked, the central heating is on, the curtains are drawn, and a gas fire is roaring in the grate when you get home" [19]. A Smart Home device is an object that becomes more functional through networking capabilities, creating a new entity. The infrastructure supporting these devices, like a base or control station, also forms part of Smart Home technology. While these devices have applications in both industrial and private settings, our focus here is on private homes [20].

3.2 IEEE 802.15.4 Standard

The IEEE 802.15.4 standard specifies the Physical (PHY) layer and Medium Access Control (MAC) sublayers. It targets wireless connections that require minimal power, suitable for devices with limited battery or specific power constraints [8]. The standard includes modes for precise distance measurement [8] and caters to devices operating globally, aiming for simple, cost-effective, and energy-efficient wireless connections among inexpensive devices, especially in the Internet of Things (IoT) [8]. Some standard versions can accurately measure distances up to about one meter [8]. It defines multiple PHY layers to operate across various frequency bands [8].

IEEE 802.15.4 is designed for Low-Rate Wireless Personal Area Networks (LR-WPANs), facilitating low-cost communication networks for wireless connectivity where power resources are limited and high data transfer rates are unnecessary [8]. The main goals of an LR-WPAN include easy setup, reliable data transmission, low cost, and good battery life [8], all while maintaining a simple and flexible protocol. An IEEE 802.15.4 network comprises two types of devices: a Full-Function Device (FFD) and a Reduced-Function Device (RFD) [8]. An FFD can act as a Personal Area Network (PAN) coordinator or a coordinator [8], whereas an RFD cannot [8]. Designed for simple applications like light switches or passive infrared sensors, RFDs do not need to transmit much data and connect to only one FFD at a time [8]. This functionality allows RFDs to be developed with fewer resources and less memory.

An LR-WPAN device includes at least one PHY layer that houses the radio frequency (RF) transceiver and its associated low-level control mechanism [8]. It also has a MAC sublayer that facilitates access to the physical channel for different types of data transmission [8]. Figure 3.1 depicts these layers. As shown in Figure 3.1, the subsequent higher layers consist of a network layer responsible for network setup, configuration, and message routing and an application layer designed to deliver the device's intended functionalities [8]. In this study, the ZigBee protocol will function in these higher layers.

3.2.1 PHY

The PHY manages the activation and deactivation of the radio transceiver, performs Energy Detection (ED), Link Quality Indication (LQI), channel selection, Clear Channel Assessment (CCA), ranging, and transmits and receives packets across the physical medium [8].

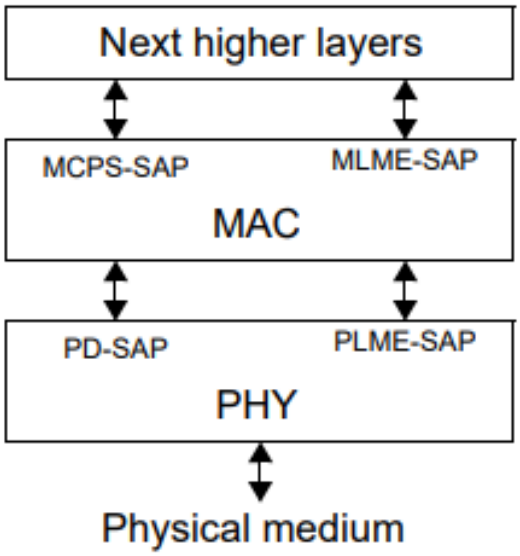

Figure 3.1: LR-WPAN device architecture [8].

3.2.2 MAC Sublayer

The MAC sublayer provides two primary services: the MAC data service and the MAC management service, interfacing with the MAC Sublayer Management Entity Service Access Point (MLME-SAP) [8]. The MAC data service enables the sending and receiving of MAC Protocol Data Units (MPDUs) through the PHY data service [8]. The MAC sublayer embeds various functionalities, including beacon management, channel access, Guaranteed Time Slot (GTS) management, frame validation, confirmation for frame delivery, association, and disassociation [8]. Furthermore, the MAC sublayer includes provisions for adding security mechanisms tailored to specific applications [8].

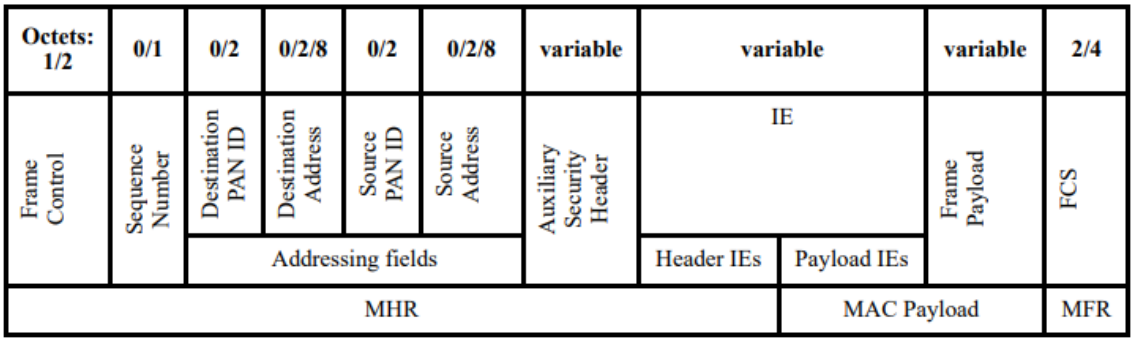

Figure 3.2: General MAC frame format [8].

3.2.3 Security Services

Ensuring the security of packets during transmission is crucial for network security. The IEEE 802.15.4 standard caters to this need by providing security services such as message integrity, confidentiality, replay protection, and access control [21]. Within the MAC layer, this standard checks the sequence and integrity of received frames before passing them to higher layers and examines frames before sending them to the PHY. This thorough process in the MAC layer ensures the application of security measures to each frame, thereby strengthening network security [8].

1. Access Control: Prevents unauthorized nodes from accessing the network and executing malicious actions. Unauthorized nodes sending fake or malicious frames are disconnected and discarded.

2. Message Integrity: Ensures that message content remains unaltered during transmission. The standard employs Message Authentication Codes (MACs) generated by shared secret cryptographic keys to authenticate messages.

3. Message Confidentiality: Aims to hide message content from unauthorized nodes through encryption and Nonce techniques, making it impossible for them to decipher the encrypted messages.

4. Replay Protection: Prevents malicious replay attacks by assigning sequence numbers to packets. Receivers verify the sequence number to accept or reject packets, protecting against resending of captured messages.

Each security service tackles specific vulnerabilities, protecting the wireless network from unauthorized access, tampering, and eavesdropping attacks [21, 22, 23].

3.3 ZigBee Protocol

This section will explore the details of the ZigBee protocol standard, focusing on its specifications, interfaces, objects, and protocols. As outlined in Chapter 1.1, the ZigBee protocol supports various network topologies, including star, tree, and mesh. In a star network, the ZigBee coordinator controls and directly manages communication with end devices, forming a centralized network. Tree and mesh networks involve the coordinator establishing the network and defining key parameters, with routers expanding the network. Tree networks utilize hierarchical routing and may use beacon-oriented communication as per IEEE 802.15.4. In contrast, mesh networks enable full peer-to-peer communication, whereas ZigBee routers do not emit regular IEEE 802.15.4 beacons [24, 25].

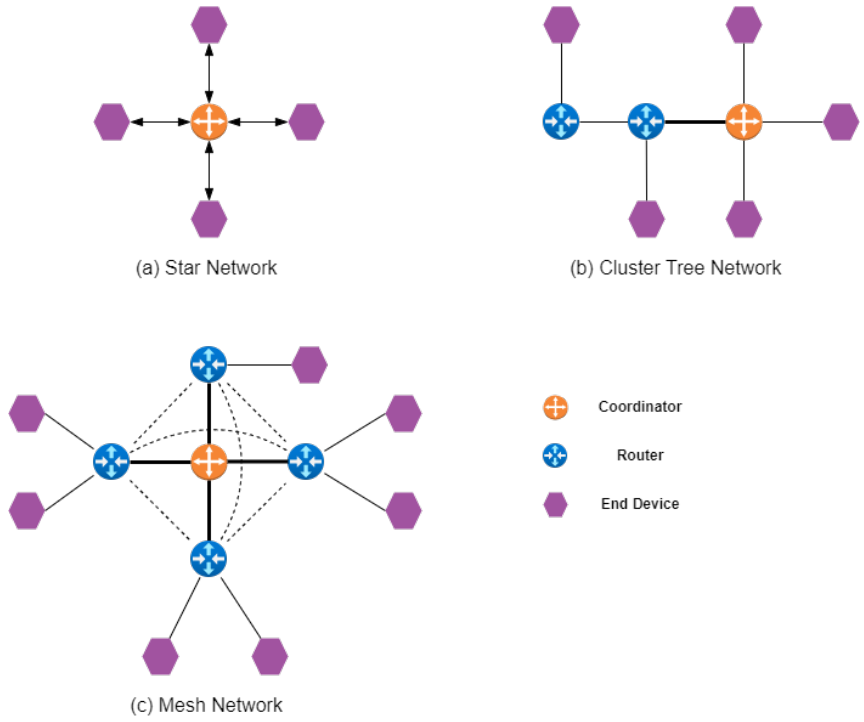

Figure 3.3: ZigBee Network Topology.

3.3.1 ZigBee Architecture Stack

The ZigBee architecture is organized into layers, each providing specific services for the one above. This protocol includes essential elements like the Application Support Sub-layer (APS), ZigBee Device Objects (ZDO), ZigBee Device Profile (ZDP), application framework, Network (NWK) layer, and security services. The Application (APL) layer's framework comprises the APS and ZDO, which support manufacturer-defined application objects and share services with the ZDO [1].

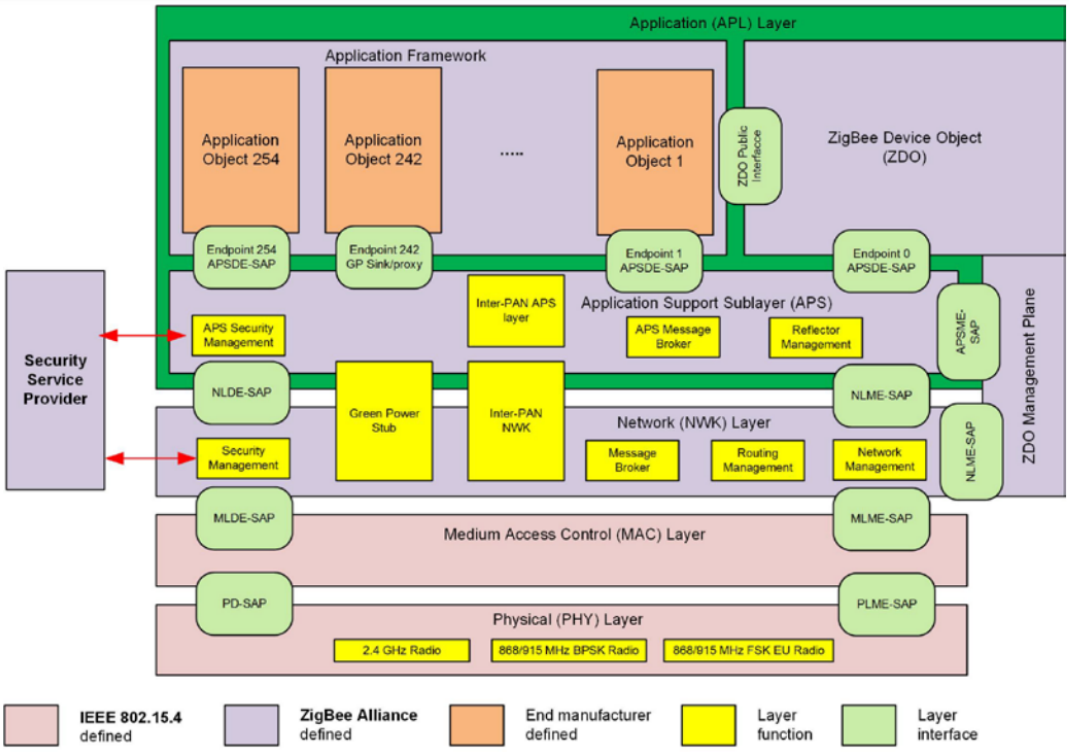

Figure 3.4: ZigBee Architecture Stack [1].

3.3.1.1 Application (APL) Layer

- **Application Support Sub-layer (APS)**
  The APS is an intermediary between the NWK and APL layers, providing various services for the ZDO and custom application objects designed by manufacturers. It offers these services through two main entities: the data and management services. The APS Data Entity (APSDE) handles data transmission using its Service Access Point (SAP), the APSDE-SAP. Meanwhile, the APS Management Entity (APSME) manages the administration services through its SAP, the APSME-SAP. Additionally, APSME maintains a set of managed objects called the APS Information Base (AIB) [1].

  The APSDE delivers a data service to the network layer, ZDO, and application objects, facilitating the exchange of application Protocol Data Units (PDU) among two or more devices within the same network. Its services include [1]:

  – Creating Application Level PDUs (APDU): Taking an application PDU and generating an APS PDU by adding necessary protocol information.

- Binding: When two devices are bound, the APSDE can transfer a message from one bound device to the other.
- Group address filtering: Filtering group-addressed messages based on endpoint group membership.
- Reliable transport: Enhancing transaction reliability beyond what's available solely from the NWK layer by using end-to-end retries.
- Rejecting duplicates: Ensuring that messages offered for transmission are not received more than once.
- Fragmentation: Allowing segmentation and reassembly of messages longer than the payload of a single NWK layer frame.

The APSME offers a management service allowing an application to engage with the stack. It enables the following services [1]:

- Binding service: Matching two devices based on their services and requirements, constructing and maintaining a table to store this data.
- Binding management: Similar to the binding service, it matches devices based on their services and needs.
- AIB management: The ability to access and modify attributes in the device's AIB.
- Security setup: Establishing secure relationships with other devices through secure keys.
- Group management: Creating a single address shared by multiple devices, adding or removing devices from the group.

<table>
<tr><td>Octets: 1</td><td>0/1</td><td>0/2</td><td>0/2</td><td>0/2</td><td>0/1</td><td>1</td><td>0/
Variable</td><td>Variable</td></tr>
<tr><td rowspan="2">Frame control</td><td>Destination endpoint</td><td>Group address</td><td>Cluster identifier</td><td>Profile identifier</td><td>Source endpoint</td><td rowspan="2">APS counter</td><td rowspan="2">Extended header</td><td rowspan="2">Frame payload</td></tr>
<tr><td colspan="5">Addressing fields</td></tr>
<tr><td colspan="8">APS header</td><td>APS pay-load</td></tr>
</table>

Figure 3.5: General APS frame format [1].

APS Command	Unique Trust Center Link Key (0x00)	Global Trust CenterLink Key (0x01)
Transport Key (0x05)	APS encryption is required as per device policy	APS encryption is required as per device policy
Update Device (0x06)	APS encryption required	APS encryption not required
Remove Device (0x07)	APS encryption required	APS encryption required
Request Key (0x08)	APS encryption required Trust Center Policy may further restrict	APS encryption required Trust Center Policy may further restrict
Switch Key (0x09)	APS encryption not required	APS encryption not required
Tunnel Data (0x0E)	APS encryption not required	APS encryption not required
Verify-Key (0x0F)	APS encryption not required.	APS encryption not required
Confirm-Key (0x10)	APS encryption required	APS encryption required.

Figure 3.6: Security Policy for Accepting APS Commands in a Centralized Security Network [1].

- **ZigBee Application Framework (ZAF)**
  In ZigBee, the application framework acts as the platform where application objects exist on ZigBee devices. It allows the creation of up to 254 distinct application objects, each identified by an endpoint address ranging from 1 to 254. Additionally, two particular endpoints, 0 and 255, are designated for specific purposes: endpoint 0 links to the ZDO for data interaction, while endpoint 255 enables broadcasting data to all application objects. The ZigBee Alliance allocates endpoints 241-254, and their usage requires approval, except for endpoint 242, which is set aside for the Green Power cluster if it is implemented [1].

- **Application Profiles**
  Application profiles, integral to the ZAF, establish agreements on messages, their formats, and operational procedures. They empower developers to create compatible, decentralized applications using application entities on various devices. These profiles facilitate applications in sending commands, requesting data, and handling command processing [1].

- **Clusters**
  Clusters, each marked by a unique cluster identifier, manage data that enters or exits the device. These identifiers maintain exclusivity within the scope of a specific application profile, assisting in effectively managing data related to various functionalities [1].

- **ZigBee Device Profile**
  The ZigBee Application Layer Specification details integrate essential features such as binding, device discovery, and service discovery within ZigBee device objects. Like other ZigBee profiles, the ZigBee Device Profile defines clusters. However, the ZigBee Device Profile clusters specify capabilities universally supported across

all ZigBee devices, unlike those in specific application profiles. The Device Profile supports four main inter-device communication functions within the ZigBee protocol: device and service discovery, end device bind, bind and unbind, binding table management, and Network management [1].

- **ZigBee Device Object**
  The ZDO is a central hub of functionality, bridging the gap between application objects, the device profile, and the APS. Positioned between the application framework and the APS, it addresses the standard requirements of all applications within a ZigBee protocol stack. The ZDO is responsible for several key tasks [1]:
  - Kickstarting the APS, the NWK layer, and the Security Service Provider (SSP).
  - Collecting setup details from end applications to organize and enforce procedures like discovery, security management, network administration, and binding management.
- **Device Discovery**
  Device discovery is how a ZigBee device locates other ZigBee devices. There are two types of device discovery requests: IEEE address requests and NWK address requests. An IEEE address request is targeted at a specific device and assumes prior knowledge of the NWK address. In contrast, an NWK address request is broadcasted and includes the known IEEE address as part of the data [1].
- **Service Discovery**
  Service discovery identifies the functions a specific device provides to other devices within the ZigBee network. This identification can be achieved by individually querying each device endpoint or utilizing a match service feature. The match service feature can broadcast the query to multiple devices or send it directly to a specific device. Descriptors are used to describe the capabilities of a device, and this information can be stored in the network if the device offering a service is not reachable at the time of the discovery request [1].

3.3.1.2 Network Layer

The network layer is a critical component responsible for ensuring the proper functioning of the IEEE 802.15.4 MAC sub-layer and establishing effective communication with the application layer. Within the network layer, two service entities interface with the application layer: the data service and the management service. The NWK layer Data Entity (NLDE) manages data transmission and is connected to its respective SAP, the NLDE-SAP. On the other hand, the NWK layer's Management Entity (NLME) oversees management operations through its associated SAP, the NLME-SAP. The NLME collaborates with the NLDE for specific management tasks and maintains a network information base (NIB) database, which contains managed objects [1].

- **Network Layer Data Entity**
  The NLDE offers a data service enabling the transfer of APDUs among two or more devices situated within the same network. Its services encompass [1]:
  - Generating Network level PDUs (NPDU): The NLDE can create an NPDU from an application support sub-layer PDU by adding a suitable protocol header.

  - Routing specific to the network structure: It can send an NPDU to the right device, which could either be the final destination or the subsequent link to the communication sequence's final destination.
  - Security measures: Ensuring a transmission's genuineness and confidentiality.

- **Network Layer Management Entity**
  The NLME offers a management service facilitating interaction between an application and the stack. Its services include [1]:

  - Configuring a new device: Setting up the stack to operate as required, such as starting as a ZigBee coordinator or joining an existing network.
  - Starting a network: Establishing a new network.
  - Network participation: Joining, rejoining, or leaving a network, including the capability for a ZigBee coordinator or router to request a device to exit the network.
  - Address assignment: ZigBee coordinators and routers assign addresses to joining devices.
  - Discovering neighbors: Identifying and storing information about nearby devices within one hop.
  - Route establishment: Finding and recording paths across the network for efficient message routing.
  - Managing reception: Controlling when a device's receiver is active and for how long, aiding MAC sub-layer synchronization or direct reception.
  - Routing methods: Employing different routing techniques (unicast, broadcast, multicast, or many-to-one) to effectively exchange data within the network.

- **NWK Layer Service**
  The NWK layer provides two services accessible through two SAPs: the NWK data service, which is accessed via the NLDE-SAP, and the NWK management service, which is accessed through the NLME-SAP. These services bridge the application and the MAC sub-layer, utilizing the MCPS-SAP and MLME-SAP interfaces. In addition to these external interfaces, there is an implicit connection between the NLME and the NLDE, enabling the NLME to utilize the NWK data service as needed [1].

- **NWK Frame Format**
  In the NWK layer, frames are composed of two primary elements [1]:

  1. The NWK header: This section includes the frame control, addressing, and sequencing data.
  2. The NWK payload: This part varies in length and carries data pertinent to the specific frame type.

  These frames are organized as fields arranged in a specific sequence. The arrangement of frame formats in this layer follows the order in which the MAC sub-layer transmits them. This transmission occurs horizontally, moving from left to right, starting with the leftmost bit. Within each field, the bits are numbered from 0 (the leftmost and least significant bit) to k-1 (the rightmost and most significant

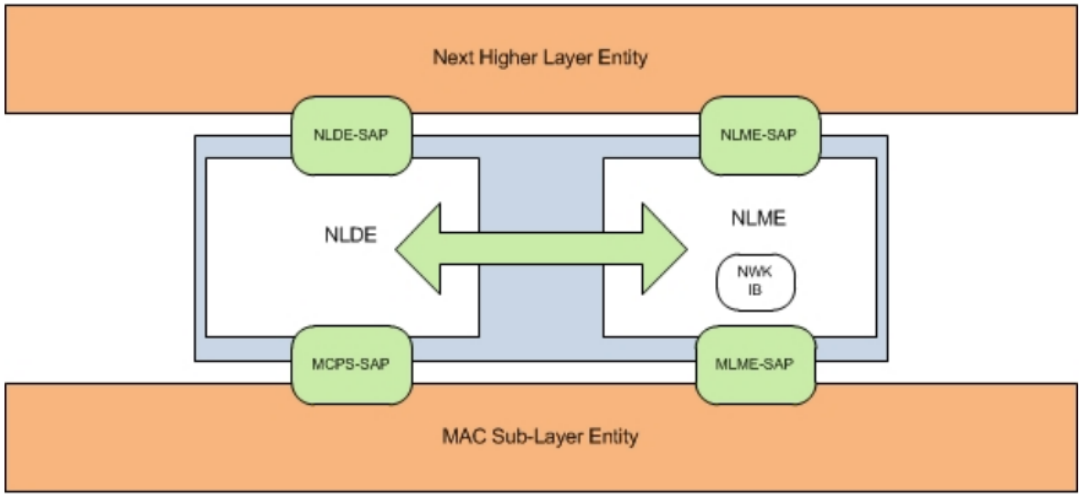

Figure 3.7: NWK layer reference model [1].

bit), with 'k' representing the total number of bits in the field. For fields that span more than one octet, they are transmitted to the MAC sub-layer in a sequence that begins with the octet containing the lowest-numbered bits and ends with the octet containing the highest-numbered bits [1].

Octets: 2	**2**	**2**	**1**	**1**	**0/8**	**0/8**	**0/1**	**Variable**	**Variable**
Frame control	Destination address	Source address	Radius	Sequence number	Destination IEEE Address	Source IEEE Address	Multicast control	Source route subframe	Frame payload
NWK Header									Payload

Figure 3.8: NWK Frame Format [1].

3.3.1.3 Security Services

ZigBee's security features encompass various mechanisms, such as key establishment, key transport, frame protection, and management of devices. These elements are integral to developing and implementing security protocols in ZigBee devices [1].

- **Security Keys**
  Encryption keys play a crucial role in maintaining the integrity and authenticity of the source and destination data. However, securing the data during transmission is essential to prevent eavesdropping and information disclosure. In ZigBee architecture, the NWK layer plays a key role in securely transmitting frames between nodes in the network. This happens by setting the NWK frame control field to 1, which indicates the use of an auxiliary frame header. The auxiliary frame includes the source address and the key sequence number of the frame. Similarly, the APS layer uses the auxiliary frame header technique for securely establishing and managing cryptographic keys [10, 1].

In ZigBee networks, security relies on two types of keys: "link" keys and a "network" key. The 128-bit link key is shared by the communicating devices and safeguards unicast communications between APL protocol peer entities. Conversely, the 128-bit network key is shared by all devices in the network and secures broadcast communications and all network layer communications. Recipients always know the security mechanism in place, whether a link or network key protects a frame. Devices obtain link keys through key-transport or pre-installation, such as during factory installation. Devices acquire network keys exclusively via key-transport. Additionally, some application profiles have developed alternative methods for generating link or network keys on devices, like out-of-band mechanisms or key negotiation protocols. Secure initiation and installation of these keys are crucial for adequate device security. The network key, versatile and singular in type, applies in both distributed and centralized security models. This model dictates the network key's distribution method and may influence the initialization of network frame counters but does not alter the method of message security. Trust center link keys fall into two categories: global and unique. The device's type determines its response to various trust center messages (APS commands), including using APS encryption. Trust center link keys may also secure APS data messages exchanged between the trust center and a peer device, with the higher layer application determining the use of APS security on these messages. A link key used between two devices, where neither is the trust center, is an application link key. In centralized security models, the default global trust center link key is set to a specific value (5A 69 67 42 65 65 41 6C 6C 69 61 6E 63 65 30 39) [1, 10, 26].

A network with enhanced security measures employs various security services. It is advisable to avoid reusing cryptographic keys across different security services to prevent potential security vulnerabilities due to unintended interactions. Consequently, these services use keys generated through a one-way function based on the link key, ensuring a distinct and logical separation in implementing various security protocols [1].

Specific purposes dictate the assignment of distinct keys: the key-load key safeguards the link keys during transport, while the key-transport key protects the network keys in transit. In the context of the ZigBee network, both the NWK and APL layers utilize the active network key. Both these layers must have access to the same network key and the corresponding incoming and outgoing frame counters. Conversely, the APS restricts the use of link keys, making them accessible solely to the APL layer [1].

The installation code is a brief code used in an algorithm to generate a 128-bit Advanced Encryption Standard (AES) key. However, this specification does not cover the specific process of key derivation from an installation code [1].

- **Trust Center Role**
  ZigBee's security framework incorporates a vital component known as the trust center, which plays a crucial role in network security and application configuration management. In each centralized security network, there is one active trust center. All network devices recognize this trust center and are responsible for creating and updating the network's security policies. On the other hand, in a distributed security network, any router can act as a trust center. These routers distribute

network security keys but do not distribute trust center link keys due to the lack of a centralized trust center [1].

Devices may already have the trust center's address and an initial trust center link key pre-installed. Alternatively, the joining device's trust center link key can be installed externally. In scenarios where temporary vulnerability is acceptable, the network key might be transmitted using APS-secured key transport with a commonly known link key [1].

In a centralized model, the trust center sets policies for new devices joining the network and oversees overall network security. This can include pre-authorizing devices before disseminating the network key or installing a preconfigured link key externally. In centralized security networks, devices securely communicate with their trust center using the current trust center link key. Trust management requires that a device only accepts a trust center link key or an active network key from its trust center through key transport [1].

For network management, devices are programmed to accept only the initial and updated network keys from their trust center, secured with the trust center link key. Regarding configuration, link keys for end-to-end security between two devices are obtained only from their trust center or through application-level negotiation using higher-level protocols. Beyond the initial trust center link or network key, all subsequent link and network keys must come from the device's trust center via secured key transport or be negotiated using higher-level application protocols [1].

- **Security Services**
  ZigBee's security architecture incorporates several built-in services to safeguard frame communication between nodes [10]:

  - Data Privacy through Symmetric Encryption: ZigBee uses a symmetric cipher to protect data privacy. Unauthorized parties without the cryptographic key cannot exploit the data.
  - AES Algorithm Implementation: ZigBee employs the AES with the Cryptographic Block Ciphers Mode (Counter with CBC-MAC) for encryption. This approach ensures both data authentication and confidentiality. ZigBee simplifies encryption by allowing the same key to be reused across different levels of its stack.
  - Data Integrity with Message Integrity Check (MIC): To safeguard data from unauthorized alterations, ZigBee uses MIC. This method also verifies the data's origin from a node possessing the cryptographic key.
  - Prevention of Forwarding Attacks: ZigBee mitigates forwarding attacks by implementing a sequential freshness counter for frame ordering (input or output). This counter resets with each new frame sent or received, allowing the detection of any malicious node that tries to interrupt and forward old frames. This mechanism prevents attackers from rerouting network traffic to malicious nodes.
  - Authentication in Network and Application Support Layers: Authentication in the NWK and APS layers is achieved using the active network key and the link key, respectively. This ensures synchronized information transfer between devices while maintaining authenticity through shared keys.

- **End Device Joining a Secured Network**
  In ZigBee networks, obtaining the network key is a critical step for all devices to become fully integrated into the network. The network coordinator manages the distribution of this key. The standard procedure involves sharing the key when a device gets authorization to join the network. As shown in Figure 3.9, the sub-protocol for this process starts with the prospective joining device sending out a beacon request to discover available networks. In response, a coordinator sends a beacon to the joiner. This beacon contains essential information, such as the coordinator's PAN ID and protocol version. The coordinator indicates its willingness to connect with new devices using the Association Permit field in the beacon [26].

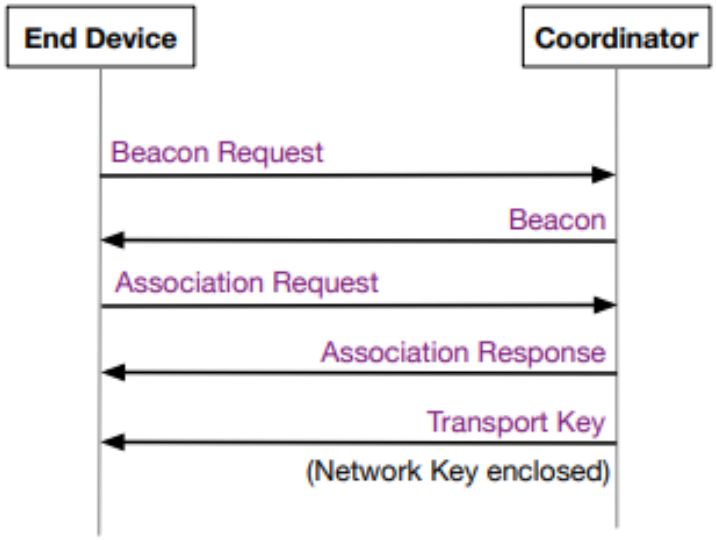

Figure 3.9: Device joining a secured network [5].

If the Association Permit value is affirmative, the joiner sends an association request to the coordinator. The coordinator then responds with an association response, which includes the Association Status field, confirming the success of the association attempt. At this stage, the device is "joined but unauthorized" [1]. Once the coordinator admits the device into the network, it transmits the encrypted network key to the device using a transport key command. The device receives this key and thus gains authorized status within the network [26].

An additional security measure in ZigBee is updating the trust center link key. After becoming part of the network, a device requests a new, unique trust center link key from the coordinator. The coordinator uses this key to encrypt all application layer messages sent to and from the device. In this sub-protocol, shown in Figure 3.10, the newly joined device requests a new trust center link key by sending a request key command to the coordinator. The coordinator then generates a device-specific trust center link key and sends it back, encrypted with the existing link key. The device acknowledges the key by sending the coordinator an APS verify key command. The coordinator completes the process by sending a confirm key command encrypted with the new trust center link key. After these exchanges, the device adopts the new trust center link key for all future communications, replacing the pre-configured link key [1, 26].

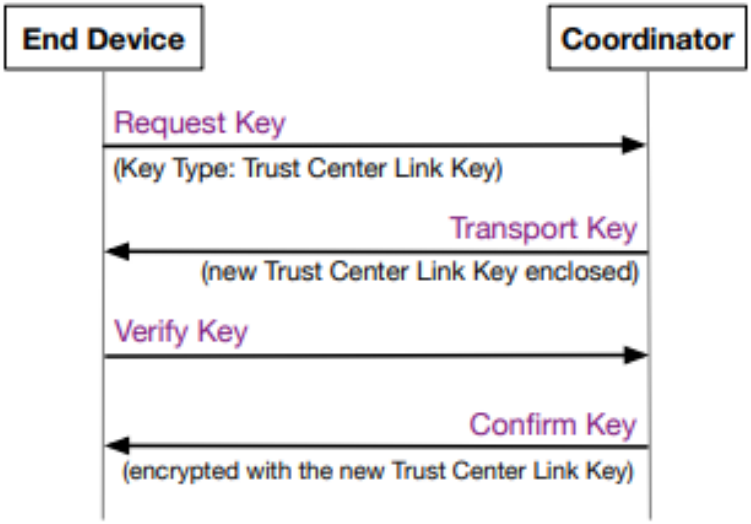

Figure 3.10: Request of a new trust center link key [26].

4 Results

This section will present the outcomes derived from the systematic literature review and the case study analysis.

4.1 Systematic Literature Review Process

Following the methodology outlined in Chapter 2.1, a total of 16 studies were chosen for inclusion in this review. The outcomes of the search process, along with the various phases of study selection, are depicted in a figure. The primary criterion for excluding studies was their irrelevance to the specific focus of this review. This typically included studies discussing ZigBee but not addressing its vulnerabilities or security attack scenarios.

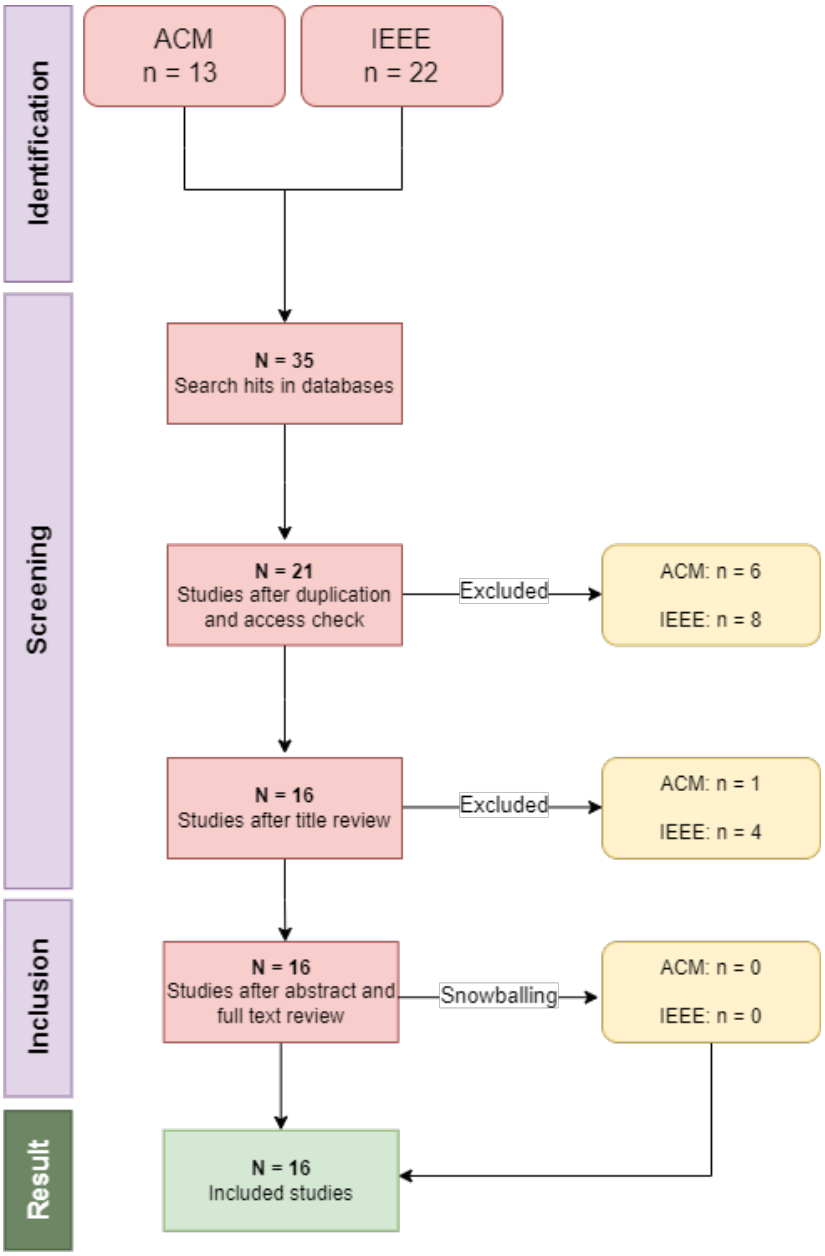

Figure 4.1: Showcases the number of studies selected at each stage of the review process. Initially, the searches resulted in 35 hits. Upon completing all the selection stages, 16 studies were ultimately chosen for inclusion. Detailed information regarding the specific studies yielded by each search is available in Table A.1, located in Appendix A.

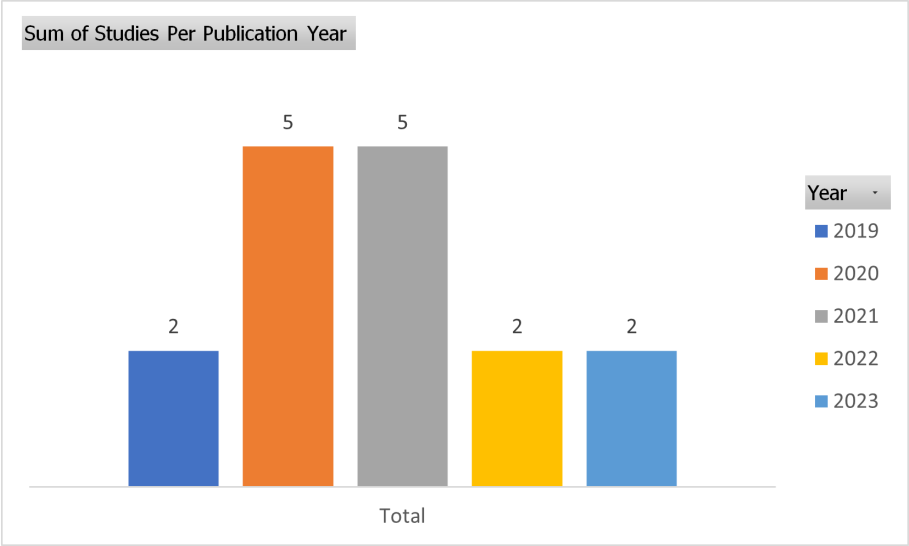

Figure 4.2: Bar chart depicting the distribution of selected studies by publication year, with a total of 16 studies ranging from 2019 to 2023.

4.2 ZigBee Vulnerabilities

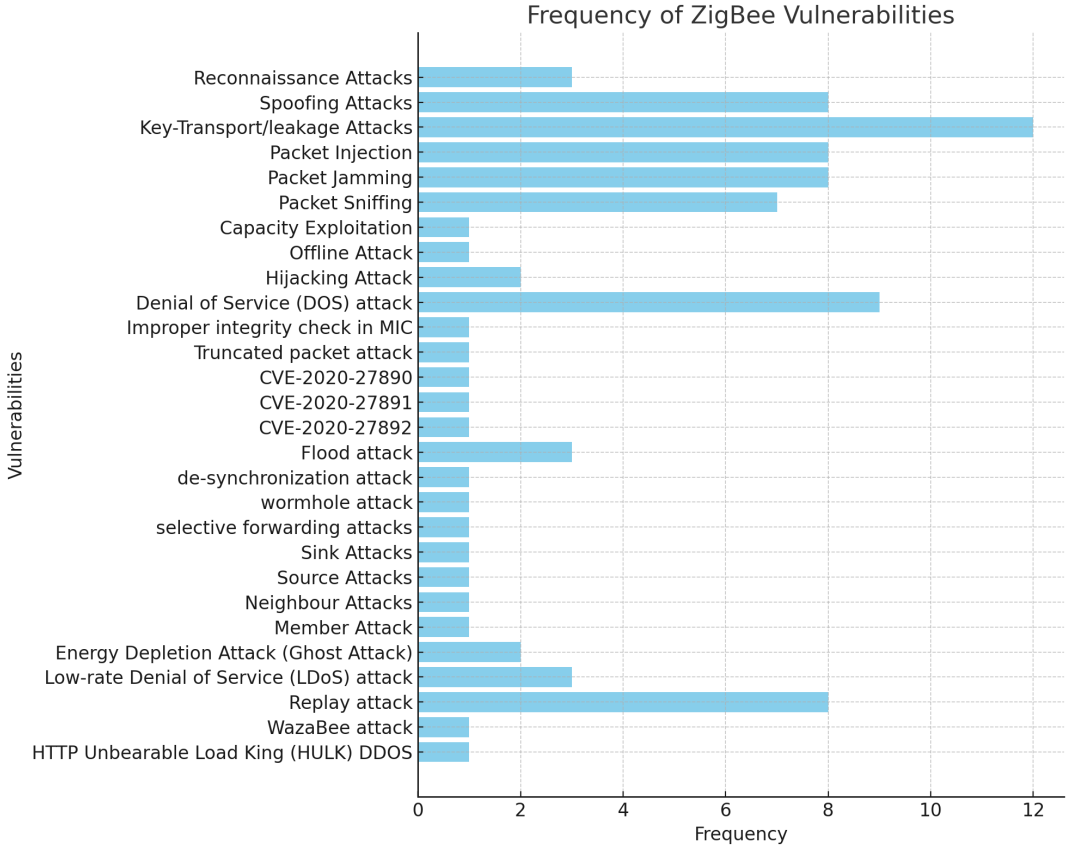

Figure 4.3: Frequency of ZigBee Network Vulnerabilities - This bar graph illustrates the frequency of 28 vulnerabilities identified in ZigBee protocol.

4.3 Non-Technical User Penetration Case

Table 4.1: Shows the evaluation of the AT by listing its software cost, installation, latest update (YY/MM/DD), exploited vulnerability, documentation, OS support, code language, and non-technical user compatibility.

Tool	Z3SEC	Zigbear	KillerBee
Software Cost	Free	Free	Free
Installation	Completed without running the program	Failed	Failed
Latest Update (YY/MM/DD)	2017/12/23	2020/02/08	2022/08/19
Exploited Vulnerabilities	None	None	None
Documentation	Clearly structured and easy to understand, but contains outdated information	Clearly structured and easy to understand	Clearly structured, difficult to understand and contains outdated information
Operating System (OS) support	Linux	Linux	Linux
Code Langauge	Python	Python	Python
Non-Technical User Compatibility	Not recommended	Not recommended	Not recommended

4.4 Technical User Penetration Case

Table 4.2: Comparative Overview of Wireshark Data Captures Detailing Packet Counts, Duration, and Dates.

Capture Number	Packets Captured	Capture Duration (seconds)	Date (YY/MM/DD)
Capture 1: All traffic	3278	491	2023/05/14
Capture 2: Philips Smart Bulb (PSB) traffic	1672	574	2023/05/14

Table 4.3: Shows a detailed description of the smart home devices by listing the PAN ID, Network Address, and 64-bit MAC Address.

Smart Home Device	PAN ID	Network Address	64-bit MAC Address
Nedis Coordinator	0x9289	0x0000	e0:79:8d:ff:fe:ac:49:8b
Aqara TaH Sensor	0x9289	0xdc61	00:15:8d:00:08:e0:28:39
Aeotec Motion Sensor	0x9289	0x6ffd	28:6d:97:00:01:12:6b:f4
PSB1	0x2e90	0x1efa	00:17:88:01:06:a1:70:5e
PSB2	0x2e90	0x0f5e	00:17:88:01:05:42:0d:6f

Table 4.4: Exploited keys with their matching description.

Exploited Key	Description
5A:69:67:42:65:65:41:6C:6C:69:61:6E:63:65:30:39	Key-transport key and key-load key
68:39:43:18:a9:06:e3:53:9a:5a:6d:fa:3f:5d:78:d0	Network key
ef:18:10:20:65:2d:22:2f:74:79:4a:0d:05:e5:5f:60	Trust center link key

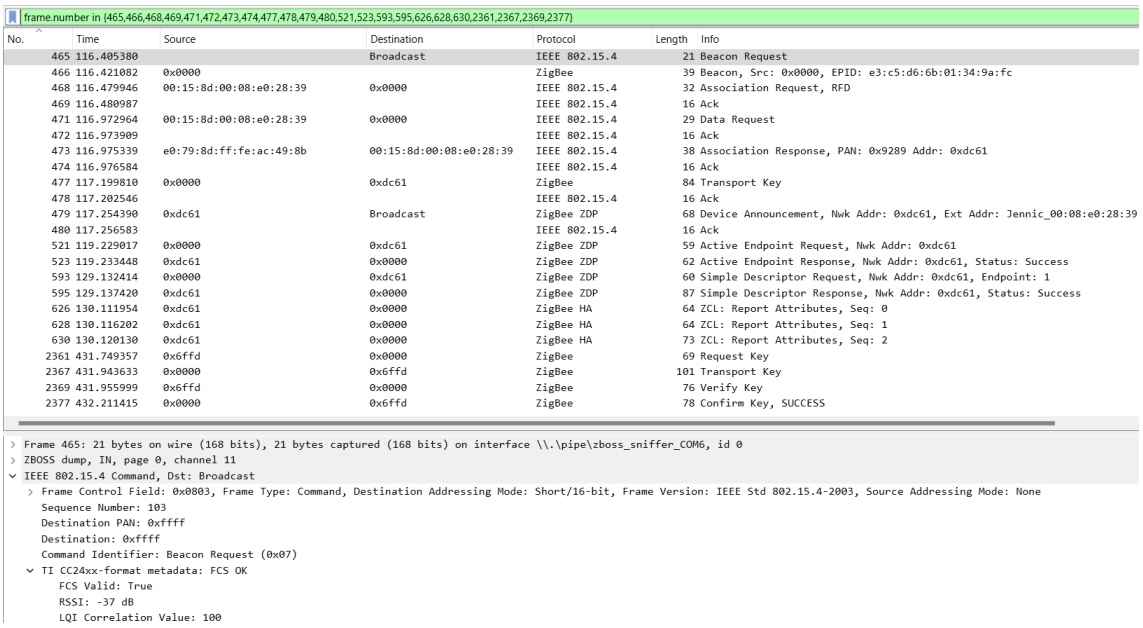

Figure 4.4: Decrypted Filtered ZigBee and IEEE 802.15.4 captured packets on Wireshark.

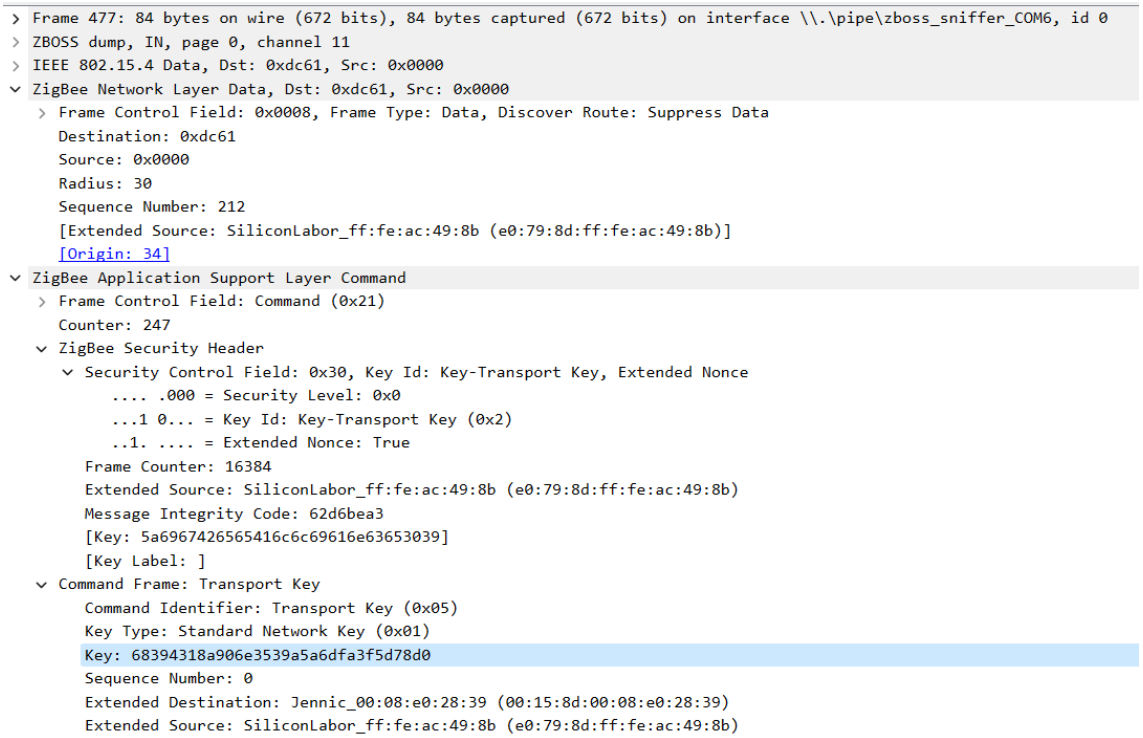

Figure 4.5: Details of packet 477 carry the decrypted key-transport key (global link key) and the standard network key.

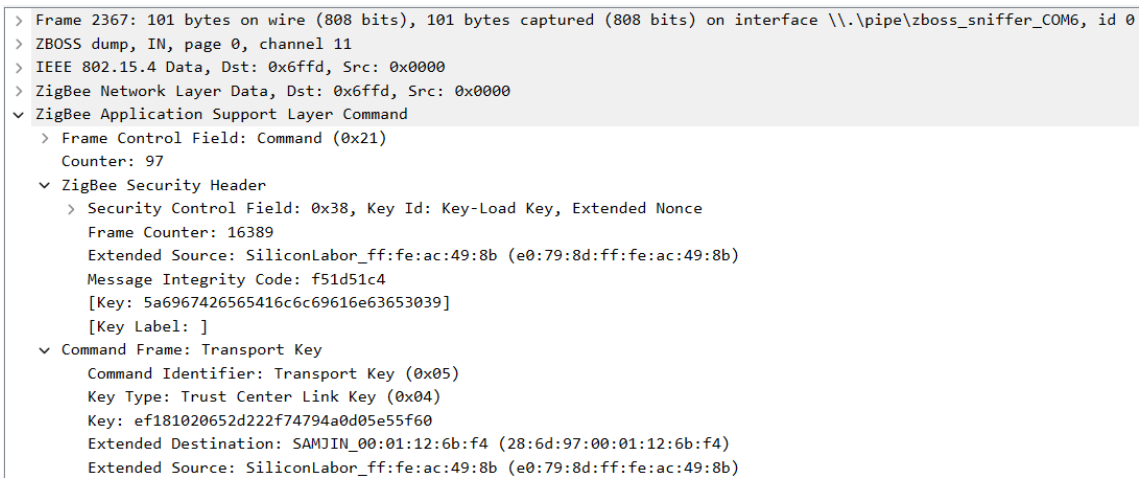

Figure 4.6: Details of packet 2367, which carries the decrypted trust center link key.

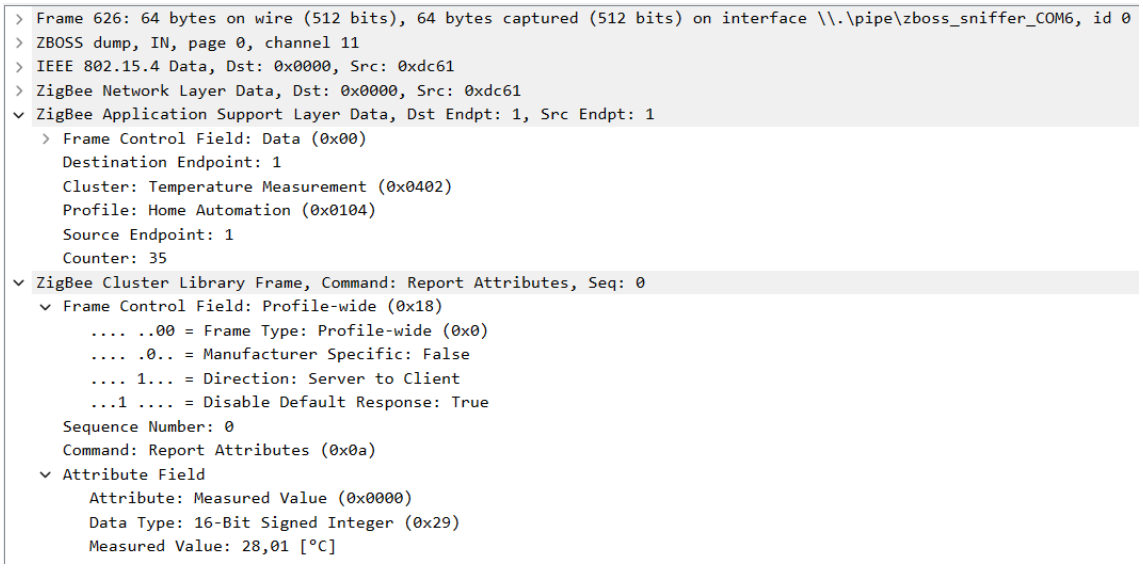

Figure 4.7: Details of packet 626, which carries the decrypted TaH sensor payload, show a temperature reading 28.01 °C.

```
> Frame 628: 64 bytes on wire (512 bits), 64 bytes captured (512 bits) on interface \\.\pipe\zboss_sniffer_COM6, id 0
> ZBOSS dump, IN, page 0, channel 11
> IEEE 802.15.4 Data, Dst: 0x0000, Src: 0xdc61
> ZigBee Network Layer Data, Dst: 0x0000, Src: 0xdc61
v ZigBee Application Support Layer Data, Dst Endpt: 1, Src Endpt: 1
  > Frame Control Field: Data (0x00)
    Destination Endpoint: 1
    Cluster: Relative Humidity Measurement (0x0405)
    Profile: Home Automation (0x0104)
    Source Endpoint: 1
    Counter: 36
v ZigBee Cluster Library Frame, Command: Report Attributes, Seq: 1
  v Frame Control Field: Profile-wide (0x18)
      .... ..00 = Frame Type: Profile-wide (0x0)
      .... .0.. = Manufacturer Specific: False
      .... 1... = Direction: Server to Client
      ...1 .... = Disable Default Response: True
    Sequence Number: 1
    Command: Report Attributes (0x0a)
  v Attribute Field
      Attribute: Measured Value (0x0000)
      Data Type: 16-Bit Unsigned Integer (0x21)
      Measured Value: 45.59 [%]
```

Figure 4.8: Details of packet 628, which carries the decrypted TaH sensor payload, show a humidity reading of 45.59%.

```
> Frame 630: 73 bytes on wire (584 bits), 73 bytes captured (584 bits) on interface \\.\pipe\zboss_sniffer_COM6, id 0
> ZBOSS dump, IN, page 0, channel 11
> IEEE 802.15.4 Data, Dst: 0x0000, Src: 0xdc61
> ZigBee Network Layer Data, Dst: 0x0000, Src: 0xdc61
v ZigBee Application Support Layer Data, Dst Endpt: 1, Src Endpt: 1
  > Frame Control Field: Data (0x00)
    Destination Endpoint: 1
    Cluster: Pressure Measurement (0x0403)
    Profile: Home Automation (0x0104)
    Source Endpoint: 1
    Counter: 37
v ZigBee Cluster Library Frame, Command: Report Attributes, Seq: 2
  v Frame Control Field: Profile-wide (0x18)
      .... ..00 = Frame Type: Profile-wide (0x0)
      .... .0.. = Manufacturer Specific: False
      .... 1... = Direction: Server to Client
      ...1 .... = Disable Default Response: True
    Sequence Number: 2
    Command: Report Attributes (0x0a)
  v Attribute Field
      Attribute: Measured Value (0x0000)
      Data Type: 16-Bit Signed Integer (0x29)
      Measured Value: 100.3 [kPa]
  v Attribute Field
      Attribute: Scale (0x0014)
      Data Type: 8-Bit Signed Integer (0x28)
      Scale: 255
  v Attribute Field
      Attribute: Scaled Value (0x0010)
      Data Type: 16-Bit Signed Integer (0x29)
      Scaled Value: 10037
```

Figure 4.9: Details of packet 630, which carries the decrypted TaH sensor payload, show a pressure reading of 100.3 kPa.

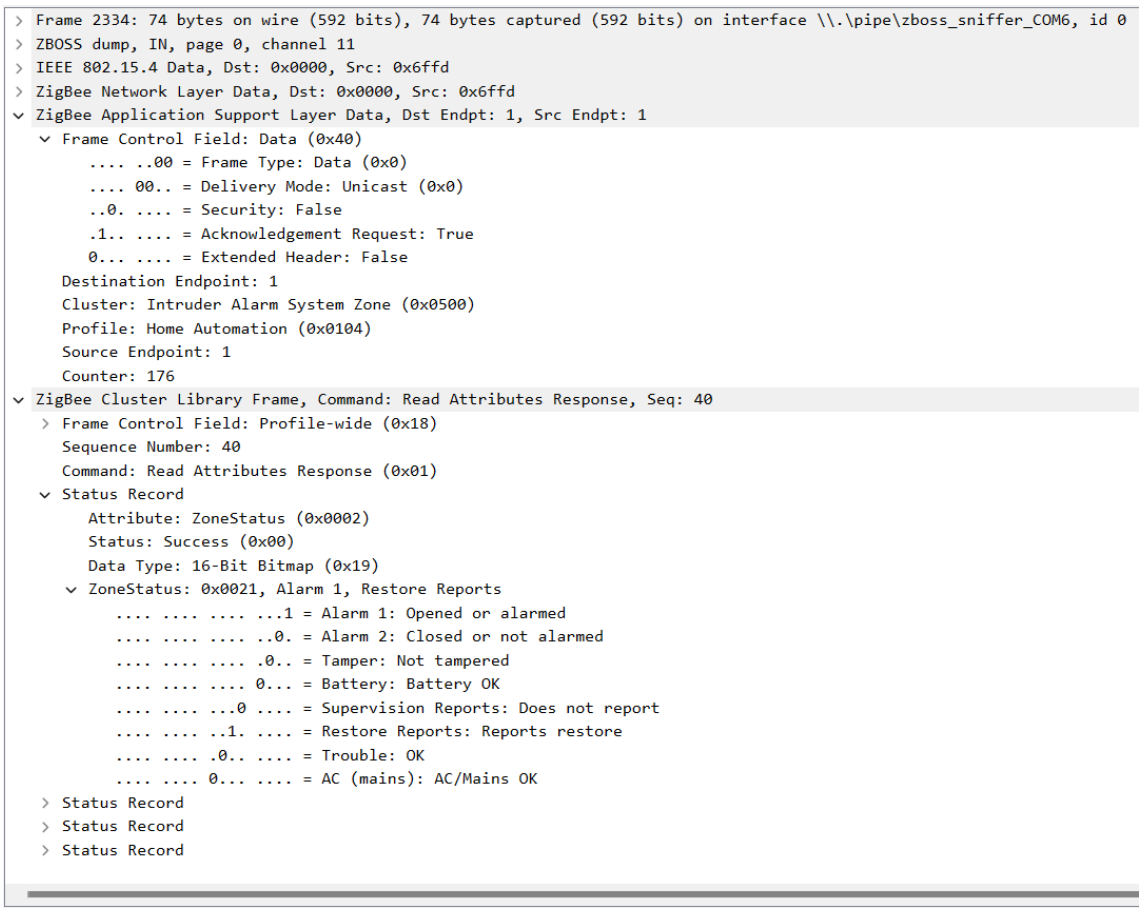

Figure 4.10: Details of packet 2334, which carries the decrypted motion sensor payload, signifying detected motion by making the "Alarm 1" field true (1).

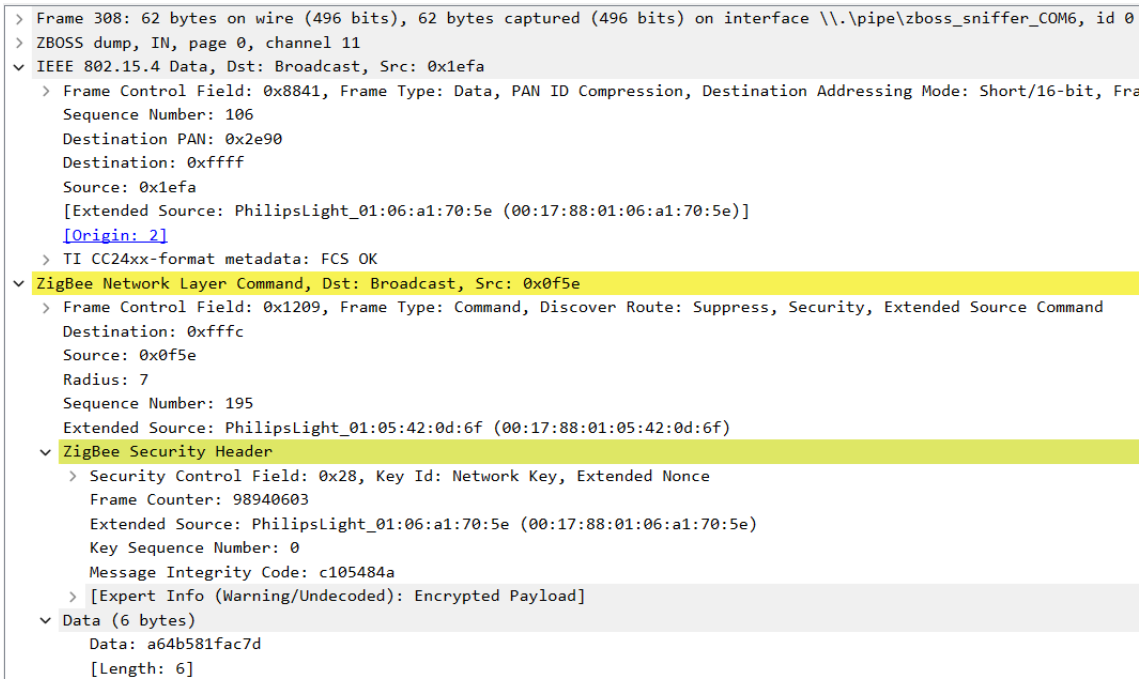

Figure 4.11: Details of packet 308, which carries the encrypted PSB1 payload.

5 Analysis

This section provides a detailed analysis of the results presented in Chapter 4

5.1 Systematic Literature Review Process

The review exclusively focused on research published between 2019 and 2023, covering studies within this time frame. Figure 4.2 visually represents the distribution of these studies across these years. Notably, 2020 and 2021 saw a significant increase in conducted studies. However, the frequency of publications in 2022 and 2023 matched that of 2019. It is essential to acknowledge that some 2023 studies might not be available for inclusion yet, given the current date of this review. Although the systematic literature review provides valuable insights, its impact suffers from including only 16 studies. The selection process was thorough, but the small research pool raises concerns about the broad applicability of our findings. The included studies commonly focus on ZigBee vulnerabilities, which could limit the generalizability of the conclusions. Caution is necessary when extending the findings to situations beyond those in the included studies. Despite these limitations, the review crucially synthesizes information, highlighting gaps for future research to bridge and enhance the broad applicability of subsequent studies. Interpreting the findings requires understanding the inherent constraints due to the currently limited state of the literature.

5.2 Research Question 1

The analysis below is based on Figure 4.3, which comprehensively visualizes the relevant data.

The frequencies range from 1 to 12, with "Key-Transport/leakage Attacks" having the highest frequency at 12. This suggests that this particular vulnerability is the most common. On the other hand, vulnerabilities like "Capacity Exploitation,"' "Offline Attack," and many others are listed with a frequency of 1, indicating that they are relatively less familiar or less frequently reported. This dataset has multiple vulnerabilities with high frequencies, such as 'Spoofing Attacks,' 'Packet Injection,' 'Packet Jamming,' and 'Denial of Service (DOS) attacks', each with a frequency of 8 or 9. These values suggest that these attacks are also significant concerns and occur relatively frequently.

To get a more concrete statistical analysis, we can calculate the descriptive statistics:

- Mean (Average): The frequency of the vulnerabilities can be calculated by summing all the frequency values and dividing by the total number of vulnerabilities listed
- Median: The middle value, when the frequencies are sorted in ascending order, can provide insight into the central tendency of the data.
- Mode: The most frequently occurring frequency value that can help identify the most common occurrence rate for these vulnerabilities.
- Standard Deviation: The standard deviation would give us an idea of how spread out the frequencies are from the mean. A high standard deviation would suggest a wide variance in the frequency of reported vulnerabilities, while a low standard deviation would indicate that the frequencies are clustered closely around the mean.
- Range: The difference between the highest and lowest frequencies.
- Variance: The average of the squared differences from the Mean.

The mean frequency of the reported vulnerabilities is approximately 3.18. This mean frequency indicates that vulnerabilities occur on average about three times within the observed dataset. The median value is 1, which tells us that half of the vulnerabilities occur once and the other half occur more than once. The mode is 1, meaning the most common frequency at which vulnerabilities occur is once. The range of frequencies is 11, indicating a wide gap between the most frequently occurring vulnerabilities and the least frequent ones. The variance is approximately 10.65, which shows considerable variability in the frequency of these vulnerabilities. The standard deviation is approximately 3.26, confirming a significant data spread. This analysis reveals that while a few vulnerabilities occur very frequently (such as 'Key-Transport/leakage Attacks' with the highest frequency of 12), many of the vulnerabilities occur only once.

5.3 Research Question 2

To answer research question 2, we analyze the technical and non-technical penetration test results.

5.3.1 Non-Technical User Penetration Case

Table 4.1 shows that all three tools are freely available, offering accessibility without financial barriers. However, the installation process varies significantly among them. Z3SEC completes installation without running the program, indicating a more straightforward setup than the other tools. In contrast, Zigbear and KillerBee fail in this aspect, posing a significant hindrance for non-technical users.

The latest update dates of the tools reflect their recency and potential maintenance level. KillerBee's most recent update in 2022 suggests active development, while Z3SEC's last update in 2017 might raise concerns about its relevance or compatibility. Regarding vulnerabilities, non-technical users have not recorded exploits with any of the tools, an essential factor for safety and security due to installation failures.

The quality of documentation is crucial for user comprehension. All tools have clearly structured documentation, but Z3SEC and KillerBee contain outdated information, and KillerBee's documentation is also challenging to understand, which could limit effective use. The documentation bias of these tools, highlighted by their clarity, comprehensibility, and recency, impacts user accessibility and effectiveness. Their complexity and outdatedness can limit usability to a specific user group, mainly those with prior technical knowledge, reducing overall accessibility and utility for non-technical users. This bias underscores the importance of inclusive, up-to-date, and user-friendly documentation in making technology tools accessible and effective for a broader range of users. Additionally, all tools exclusively support the Linux OS, limiting their user base to those familiar with this operating system, a significant limitation in terms of broader accessibility. The choice of Python as the programming language for all tools suggests ease of use for those familiar with it, but it does not benefit non-technical users. Moreover, none of the tools are recommended for non-technical users, highlighting a significant gap in user-friendliness and general accessibility.

This analysis reveals that while these tools have free and structured documentation advantages, they also face notable limitations, especially in non-technical user compatibility and documentation. There is a marked need for improvements in user interfaces, simplified usage processes, and updated documentation to make these tools more accessible to a broader audience. The inconsistency in maintenance, evidenced by the varying dates of the latest updates, also highlights the need for regular updates for tool relevance and security. It is crucial to avoid bias and not overstate the tools' capabilities

based on the available data. For instance, while KillerBee's recent update might suggest superiority, it does not necessarily correlate with more effective or user-friendly features. The analysis should remain objective, focusing on the data without inferring unverified capabilities. If quantitative data, like user satisfaction ratings or performance metrics, were available, statistical testing would provide a stricter evaluation, backing claims with empirical evidence and thus strengthening the analysis.

5.3.2 Technical User Penetration Case

Table 4.2 shows two Wireshark captures recorded. Capture 1 displays traffic from all the smart home devices listed in Table 4.3, while capture 2 shows traffic from PSB1, PSB2, and the coordinator. From capture 1, we extracted Figures 4.4 - 4.10, and from capture 2, we extracted Figure 4.11. In capture 1, we filtered 3278 packets down to 20 packets visible in Figure 4.4 to explain the compromise of the keys in Table 4.4. Since the global trust center link key is publicly available, we used it in Wireshark to decrypt the traffic. This decryption attempt succeeded, as illustrated in Figure 4.4. Figure E.1 displays an encrypted version of the duplicate packets.

In capture 1, packet 465 shows the end device broadcasting a beacon request to find any coordinators within its transmission range. The coordinator then listens to the beacon request and responds with a beacon containing information like the extended PAN ID, stack profile, and protocol version. The coordinator allows the end device to join the network using the Association Permit field in the beacon. As packet 468 shows, the end device sends an association request, revealing its 64-bit MAC address and the PAN ID from the coordinator. The Aqara TaH sensor corresponds to this 64-bit MAC address. Therefore, in packet 471, the sensor tries to pull the association response from the coordinator using the data request command. The coordinator then responds with an association response, indicating a successful association. Further, in packet 477, the coordinator sends the network key to the TaH sensor using the transport key command to secure subsequent communication. The newly joined TaH sensor broadcasts a device announcement command with its network address to avoid duplication with other devices. After the announcement, the coordinator uses active endpoint and simple descriptor request commands to understand the TaH sensor's services and create an application profile. Packets 626, 628, and 630, displayed in Figures 4.7, 4.8, and 4.9, respectively, show the TaH sensor's reported attributes. The Aeotec motion sensor undergoes the same process as the TaH sensor, and its reported attributes appear in Figure 4.10.

Figures 4.5 and 4.6 reveal that the global trust center link key functions as both a key-transport key and a key-load key. This key encrypts the standard network key and the trusted center link key, exposing the unique network key for PAN ID 0x9289 and the trusted center link key. The network key leakage presents significant security risks, such as eavesdropping, data tampering, device impersonation, replay attacks, DoS attacks, and Man-in-the-Middle (MitM) attacks.

The exposure of the trusted center link key is also concerning. This key encrypts some APS commands listed in Table 3.6, as seen in packets 2361, 2367, 2369, and 2377, particularly for commands like removing a device, requesting a key, and confirming a key. Risks include device impersonation, compromised data integrity, and key leakage.

These vulnerabilities mainly stem from manufacturers' inadequate security measures. During factory installation, they use the global trust center link key as a key-transport key for network key acquisition. This applies to all smart home devices in Table 4.3, except PSB1 and PSB2. The Nedis system's lack of a security level selection option implies non-utilization of ZigBee protocol's maximum security services. Our observations and

analysis indicate that the Nedis coordinator does not require end devices to use the newly generated trust center link key, continuing to encrypt traffic within the APS sublayer with the global trust center link key.

An in-depth examination of capture 1 revealed an unusual aspect of PSBs' communication. Figure 4.11 hypothesizes that the coordinator might have allocated a new PAN ID for the PSBs, indicated by destination PAN ID 0x2e90 in the IEEE 802.15.4 frame. To focus on PSB traffic, we specifically undertook capture 2. We tried decrypting the traffic using the global trust center link key in Wireshark, but this was unsuccessful, leaving the traffic encrypted as shown in Figure 4.11. This suggests the 0x2e90 network likely uses a unique, pre-configured link key set by Philips to avoid earlier mentioned security vulnerabilities. Since we could not decrypt the traffic in capture 2, we cannot draw definitive conclusions about PSBs' operations. Any claims at this point would be speculative.

6 Discussion

This section answers RQ3 and discusses the related work and findings.

6.1 Research Question 3

The future of ZigBee networks appears to be heading towards an increased focus on security and user accessibility. Given the vulnerabilities identified, such as Key-Transport/leakage Attacks, Spoofing Attacks, Packet Injection, and Denial of Service attacks, these networks need enhanced security measures. The systematic literature review indicates growing research interest in ZigBee, especially in addressing these vulnerabilities.

For effective deployment, the best environments for ZigBee networks, given their current vulnerabilities, are those where security can be meticulously managed. This includes controlled industrial settings and smart home environments where the advantages of ZigBee's low power consumption and efficient mesh networking can be maximized while minimizing risks. In these settings, deployers must purchase ZigBee devices that allow for manual installation and offer a choice of security levels from trusted manufacturers. This approach ensures that deployers have greater control over the security of their ZigBee network, enabling them to tailor it to their specific needs.

Furthermore, there is a need for a shift in the business approach of ZigBee product manufacturers. This change should focus on increased collaboration among manufacturers to implement more stringent security mechanisms, especially for the pre-configured global trusted link key. Such collaboration could develop more secure and standardized security protocols across different ZigBee devices, significantly reducing the risk of common vulnerabilities.

In summary, while ZigBee networks present a promising future for IoT applications, their success and safety hinge on addressing security vulnerabilities and improving deployment practices. This includes carefully selecting devices from trusted manufacturers, manual installation options, and a collaborative effort by manufacturers to enhance pre-configured security features. Controlled environments where these security measures can be effectively applied and monitored are currently the most suitable for deploying ZigBee networks.

6.2 Related Work and Findings

The findings of this study show similarities with related work regarding the key leakage vulnerability mentioned in Chapters 5.2 and 5.3 [10]. Furthermore, KillerBee was evaluated as an open-source tool in one of the studies in the related work [27]. The most significant limitation affecting this study is the lack of time. With more time available, we could include subjects for non-technical and technical user analyses without involving the researchers. Additionally, there would be the potential to analyze the PSBs' security mechanisms to understand the varied implementations of the ZigBee protocol by different manufacturers. While the findings are insightful, they are specific to the hardware, software tools, and vulnerabilities analyzed. Generalizing these results to all ZigBee networks or IoT devices may not be accurate without further research. The study's specific context is essential when applying these findings to other settings.

7 Conclusion and Future Work

This section draws the conclusions from the results and analysis. It also discusses possible future work.

7.1 Conclusion

In terms of ZigBee vulnerabilities, the analysis identified "Key-Transport/leakage Attacks" as the most common vulnerability, alongside others like "Spoofing Attacks," "Packet Injection," and "Denial of Service (DOS) attacks." The descriptive statistical analysis revealed a mean frequency of vulnerabilities around 3.18, with a wide range of frequencies indicating a significant variability in the occurrence of these vulnerabilities. This finding underscores the need for ongoing attention to security in ZigBee networks.

For non-technical users, tools such as Z3SEC, Zigbear, and KillerBee, despite being free, need to be improved in installation and documentation. KillerBee's recent update in 2022 suggests active development, contrasting with Z3SEC's last update 2017. The documentation for these tools is generally clear but can be outdated or complex, limiting their accessibility to non-technical users. All tools support only the Linux OS and are written in Python, further narrowing their user base. Consequently, non-technical users can not exploit the ZigBee network in any way.

In the technical user penetration case, Wireshark captures of ZigBee network traffic revealed considerable security risks, primarily due to the exposure of critical keys like the global trust center link and network key. This vulnerability exposes the network to attacks, including eavesdropping, data tampering, and device impersonation. These security issues primarily arise from inadequate security measures implemented by manufacturers and the non-utilization of ZigBee's maximum security services. An interesting observation was made regarding Philips' devices, which employ unique security measures. However, how PSBs implement ZigBee security services remains beyond the purview of this current investigation and needs further exploration. This situation underscores the importance of robust security practices in deploying and maintaining ZigBee networks.

This study has successfully addressed the knowledge gap by demonstrating that non-technical users cannot exploit the ZigBee smart home network, whereas technical users possess this capability. This finding is of significant societal relevance, as it reassures individuals who are considering installing ZigBee smart home automation systems. Due to the complexities involved, they can be assured of a reduced risk of unauthorized access or hacking by non-technical users. A comprehensive understanding of the ZigBee protocol and other pertinent aspects of CS is requisite to compromise these automation systems, thus limiting the potential for exploitation to individuals with advanced technical skills.

7.2 Future Work

In the discussion, it was noted that time constraints significantly limited this study. Another critical limitation was the need for more funding. With more financial support, the study could have included a more comprehensive range of ZigBee devices and enhanced sniffing hardware. This expansion would have allowed for more extensive testing of different attack types. Additionally, with a broader selection of ZigBee devices, the study could have explored compatibility issues among devices from various manufacturers. More time would have enabled a deeper investigation into the ZigBee protocol, thereby improving the insights gained in the technical user penetration case. Considering the constraints in the systematic literature review, addressing this limitation

could be a focal point for future work. A more extensive literature review, facilitated by additional resources, could provide a more comprehensive understanding of the existing research landscape. This could involve exploring publications in related domains, considering alternative databases, or employing advanced search strategies to uncover hidden insights. By expanding the scope of the literature review, future research can contribute to filling the gaps identified in the current study, offering a more nuanced perspective on the subject matter.

References

[1] Zigbee Alliance, "zigbee specification," https://csa-iot.org/wp-content/uploads/2022/01/docs-05-3474-22-0csg-zigbee-specification-1.pdf, Apr. 2017, accessed: 2023-11-20.

[2] "What is the internet of things (IoT)?" https://www.oracle.com/internet-of-things/what-is-iot/, accessed: 2023-4-17.

[3] H. Allioui and Y. Mourdi, "Exploring the full potentials of iot for better financial growth and stability: A comprehensive survey," *Sensors*, vol. 23, no. 19, p. 8015, 2023.

[4] "BBC STORYWORKS," https://www.bbc.com/storyworks/future/the-new-wave/innovation, accessed: 2023-4-17.

[5] MiSecurity, "Cyber security baseline for consumer internet of things device," https://github.com/MiSecurity/Cyber-Security-Baseline-for-Consumer-Internet-of-Things/blob/main/resources/pdf/Cyber%20Security%20Baseline%20for%20Consumer%20Internet%20of%20Things%20Device.pdf., Feb. 2021, accessed: 2023-4-17.

[6] R. Li, W. Zhang, L. Wu, Y. Tang, and X. Xie, "ZPA: A smart home privacy analysis system based on ZigBee encrypted traffic," *Wirel. Commun. Mob. Comput.*, vol. 2023, pp. 1–16, 2023.

[7] J. Li, X. Zhu, N. Tang, and J. Sui, "Study on ZigBee network architecture and routing algorithm," in *2010 2nd International Conference on Signal Processing Systems*. IEEE, 2010.

[8] "Ieee standard for low-rate wireless networks," *IEEE Std 802.15.4-2020 (Revision of IEEE Std 802.15.4-2015)*, pp. 1–800, 2020.

[9] A. Allakany, A. Saber, S. M. Mostafa, M. Alsabaan, M. I. Ibrahem, and H. Elwahsh, "Enhancing security in zigbee wireless sensor networks: A new approach and mutual authentication scheme for d2d communication," *Sensors*, vol. 23, no. 12, p. 5703, 2023.

[10] S. Khanji, F. Iqbal, and P. Hung, "Zigbee security vulnerabilities: Exploration and evaluating," in *2019 10th international conference on information and communication systems (ICICS)*. IEEE, 2019, pp. 52–57.

[11] M. Praveen, A. Raza, and M. Hasib, "Open-source security testing tools for iot protocols-mqtt and zigbee," in *2023 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2023, pp. 01–06.

[12] M. A. Bin Karnain and Z. Bin Zakaria, "A review on zigbee security enhancement in smart home environment," in *2015 2nd International Conference on Information Science and Security (ICISS)*, 2015, pp. 1–4.

[13] Y. Al-Hadhrami and F. K. Hussain, "DDoS attacks in IoT networks: a comprehensive systematic literature review," *World Wide Web*, vol. 24, no. 3, pp. 971–1001, 2021.

[14] B. Kitchenham and S. Charters, *Guidelines for performing Systematic Literature Reviews in Software Engineering*, Durham, UK, Technical, 2007.

[15] Y. Xiao and M. Watson, "Guidance on conducting a systematic literature review," *Journal of Planning Education and Research*, vol. 39, no. 1, pp. 93–112, 2019. [Online]. Available: https://doi.org/10.1177/0739456X17723971

[16] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering*, ser. EASE '14. New York, NY, USA: Association for Computing Machinery, 2014. [Online]. Available: https://doi.org/10.1145/2601248.2601268

[17] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empir. Softw. Eng.*, vol. 14, no. 2, pp. 131–164, 2009.

[18] Texas Instruments Incorporated, "CC debugger user's guide," https://www.ti.com/lit/ug/swru197h/swru197h.pdf, accessed: 2023-8-17.

[19] "Oxford languages," http://oxforddictionaries.com/definition/english/smart-home, Sep. 2019, accessed: 2023-11-20.

[20] M. Schiefer, "Smart home definition and security threats," in *2015 Ninth International Conference on IT Security Incident Management & IT Forensics*. IEEE, 2015.

[21] A. A. ALABDULATIF, "Analyse security of the disassociation procedure in the ieee 802.15. 4 standard," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 18, no. 5, 2020.

[22] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet of things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.

[23] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in internet of things: The road ahead," *Computer networks*, vol. 76, pp. 146–164, 2015.

[24] T. Kumar and P. B. Mane, "Zigbee topology: A survey," in *2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 2016, pp. 164–166.

[25] O. G. Aju, "A survey of zigbee wireless sensor network technology: Topology, applications and challenges," *International Journal of Computer Applications*, vol. 130, no. 9, pp. 47–55, 2015.

[26] L. Li, P. Podder, and E. Hoque, "A formal security analysis of zigbee (1.0 and 3.0)," in *Proceedings of the 7th Symposium on Hot Topics in the Science of Security*, 2020, pp. 1–11.

[27] H. Chen, C. Meng, Z. Shan, Z. Fu, and B. K. Bhargava, "A novel low-rate denial of service attack detection approach in ZigBee wireless sensor network by combining Hilbert-Huang transformation and trust evaluation," *IEEE Access*, vol. 7, pp. 32 853–32 866, 2019.

[28] M. S. Wara and Q. Yu, "New replay attacks on ZigBee devices for internet-of-things (IoT) applications," in *2020 IEEE International Conference on Embedded Software and Systems (ICESS)*. IEEE, 2020.

[29] R. Cayre, F. Galtier, G. Auriol, V. Nicomette, M. Kaaniche, and G. Marconato, "WazaBee: attacking zigbee networks by diverting bluetooth low energy chips," in *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2021.

[30] S. Okada, D. Miyamoto, Y. Sekiya, and H. Nakamura, "New LDoS attack in zigbee network and its possible countermeasures," in *2021 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2021.

[31] E. A. Asonye, I. Anwuna, and S. M. Musa, "Securing ZigBee IoT network against HULK distributed denial of service attack," in *2020 IEEE 17th International Conference on Smart Communities: Improving Quality of Life Using ICT, IoT and AI (HONET)*. IEEE, 2020.

[32] S. Okada, D. Miyamoto, Y. Sekiya, H. Takase, and H. Nakamura, "LDoS attacker detection algorithms in zigbee network," in *2021 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics)*. IEEE, 2021.

[33] O. Lourme, G. Grimaud, and M. Hauspie, "Zbds2023: A multi location zigbee dataset to build innovative iot intrusion detection systems," in *2023 19th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2023, pp. 84–91.

[34] H. Pirayesh, P. Kheirkhah Sangdeh, and H. Zeng, "Securing ZigBee communications against constant jamming attack using neural network," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4957–4968, 2021.

[35] D.-G. Akestoridis, M. Harishankar, M. Weber, and P. Tague, "Zigator: Analyzing the security of zigbee-enabled smart homes," in *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 77–88. [Online]. Available: https://doi.org/10.1145/3395351.3399363

[36] J. Wang, Z. Li, M. Sun, and J. C. Lui, "Zigbee's network rejoin procedure for iot systems: Vulnerabilities and implications," in *Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses*, ser. RAID '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 292–307. [Online]. Available: https://doi.org/10.1145/3545948.3545953

[37] W. Wang, F. Cicala, S. R. Hussain, E. Bertino, and N. Li, "Analyzing the attack landscape of zigbee-enabled iot systems and reinstating users' privacy," in *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 133–143. [Online]. Available: https://doi.org/10.1145/3395351.3399349

[38] X. Wang and S. Hao, "Don't kick over the beehive: Attacks and security analysis on zigbee," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 2857–2870. [Online]. Available: https://doi.org/10.1145/3548606.3560703

[39] M. Ren, X. Ren, H. Feng, J. Ming, and Y. Lei, "Z-fuzzer: Device-agnostic fuzzing of zigbee protocol implementation," in *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 347–358. [Online]. Available: https://doi.org/10.1145/3448300.3468296

[40] "Flash-programmer," https://www.ti.com/tool/FLASH-PROGRAMMER, accessed: 2023-8-16.

[41] "killerbee: IEEE 802.15.4/ZigBee security research toolkit," https://github.com/riverloopsec/killerbee, accessed: 2023-8-16.

[42] "DSR IoT," https://dsr-iot.com/downloads/tools, accessed: 2023-8-16.

[43] IoTsec, "Z3sec: Penetration testing framework for ZigBee security research," https://github.com/IoTsec/Z3sec, accessed: 2023-8-16.

[44] D. Cauquil, "cc2531-killerbee-fw: Killerbee compatible ZigBee sniffer/injector firmware for TI CC2531 USB dongles," https://github.com/virtualabs/cc2531-killerbee-fw, accessed: 2023-8-16.

[45] P. Normann, "zigbear: Zigbee security research toolkit for the RaspBee, nRF52840 and CC2531 radio modules," https://github.com/philippnormann/zigbear, accessed: 2023-8-16.

Appendices

A Systematic Literature Review Results

Table A.1: List of Selected Studies from IEEE and ACM Databases.

Database	Date (YY/MM/DD)	Selected Studies	Publication
IEEE	2023/06/23	[10], [28], [29], [30], [31], [32], [11], [33]	Conference
IEEE	2023/06/23	[34], [27]	Journal
ACM	2023/06/25	[35], [36], [26], [37], [38], [39]	Conference

B Hardware and Software Tools

Table B.1: Shows the hardware and software used in this case study with its corresponding description.

Hardware	Description
2 x Philips hue A60 E27	Warm white smart light bulb - uses ZigBee 3.0 protocol.
Personal Computer (PC)	Uses Windows 11 pro operating system (OS) to conduct the penetration test.
Aqara TaH Sensor	Uses ZigBee protocol.
CC2531 USB stick	Is used to sniff ZigBee packets.
Texas Instruments CC debugger	Is used to flash the CC2531 USB Stick.
CC2531 downloader cable	Is used to flash the CC2531 USB Stick.
Nedis ZigBee 3.0 gateway	Acts as the coordinator of the network and is accessed through the Nedis SmartLife Application (NSLA).
Aeotec motion sensor	Uses ZigBee 3.0 protocol
Lenovo ideapad L340 laptop	Is used to run Ubuntu 23.10 desktop.
Software	**Description**
SmartRF Flash Programmer ver.1.12.8	Is used to program the flash memory in Texas Instruments [40].
KillerBee	KillerBee is a Framework and Tool for Testing and Auditing ZigBee and IEEE 802.15.4 Networks [41].
ZBOSS sniffer	Is an open-source cross-platform ZigBee packet sniffer [42].
Z3SEC	Is an open-source penetration testing framework comprising a set of command-line tools for ZigBee security research [43].
Bumblebee	Is a specific firmware for CC2531 that allows ZigBee packet sniffing and injection based on KillerBee [44].
Ubuntu 23.10	Is used to run KillerBee, BumbleBee, Z3SEC, and Zigbear.
NSLA	The application used to control the Nedis gateway.
Wireshark	Is used to analyze the ZigBee packets captured.
Zigbear	A Zigbee security research toolkit for the RaspBee, nRF52840 and CC2531 radio modules. [45].
AT	Zigbear, Bumblebee, KillerBee, and Z3SEC .

C Hash Values

Table C.1: SHA-256 hash value of the collected data.

Collected Data	SHA-256 Hash Value
Capture 1.pcapng	5f720ccdbcc4f996cc870c0f1c39e2d0e01a31876d86eebef5d2582ad705da0e
Capture 2.pcapng	dfda7d64e348263f92af15476414e9530e1a0ff9d9622c62b3461777339161d0

D Installation of AT

```
creating build/bdist.linux-x86_64/egg/EGG-INFO
installing scripts to build/bdist.linux-x86_64/egg/EGG-INFO/scripts
running install_scripts
running build_scripts
creating build/scripts-3.11
copying and adjusting tools/z3sec_touchlink -> build/scripts-3.11
copying and adjusting tools/z3sec_control -> build/scripts-3.11
copying and adjusting tools/z3sec_key_extract -> build/scripts-3.11
copying and adjusting tools/z3sec_show -> build/scripts-3.11
copying and adjusting tools/z3sec_install_code -> build/scripts-3.11
changing mode of build/scripts-3.11/z3sec_touchlink from 644 to 755
changing mode of build/scripts-3.11/z3sec_control from 644 to 755
changing mode of build/scripts-3.11/z3sec_key_extract from 644 to 755
changing mode of build/scripts-3.11/z3sec_show from 644 to 755
changing mode of build/scripts-3.11/z3sec_install_code from 644 to 755
creating build/bdist.linux-x86_64/egg/EGG-INFO/scripts
copying build/scripts-3.11/z3sec_install_code -> build/bdist.linux-x86_64/egg/EGG-INFO/scripts
copying build/scripts-3.11/z3sec_key_extract -> build/bdist.linux-x86_64/egg/EGG-INFO/scripts
copying build/scripts-3.11/z3sec_touchlink -> build/bdist.linux-x86_64/egg/EGG-INFO/scripts
copying build/scripts-3.11/z3sec_control -> build/bdist.linux-x86_64/egg/EGG-INFO/scripts
copying build/scripts-3.11/z3sec_show -> build/bdist.linux-x86_64/egg/EGG-INFO/scripts
changing mode of build/bdist.linux-x86_64/egg/EGG-INFO/scripts/z3sec_install_code to 755
changing mode of build/bdist.linux-x86_64/egg/EGG-INFO/scripts/z3sec_key_extract to 755
changing mode of build/bdist.linux-x86_64/egg/EGG-INFO/scripts/z3sec_touchlink to 755
changing mode of build/bdist.linux-x86_64/egg/EGG-INFO/scripts/z3sec_control to 755
changing mode of build/bdist.linux-x86_64/egg/EGG-INFO/scripts/z3sec_show to 755
copying z3sec.egg-info/PKG-INFO -> build/bdist.linux-x86_64/egg/EGG-INFO
copying z3sec.egg-info/SOURCES.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying z3sec.egg-info/dependency_links.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying z3sec.egg-info/top_level.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
zip_safe flag not set; analyzing archive contents...
creating dist
creating 'dist/z3sec-0.1-py3.11.egg' and adding 'build/bdist.linux-x86_64/egg' to it
removing 'build/bdist.linux-x86_64/egg' (and everything under it)
Processing z3sec-0.1-py3.11.egg
Copying z3sec-0.1-py3.11.egg to /usr/local/lib/python3.11/dist-packages
Adding z3sec 0.1 to easy-install.pth file
z3sec_control:160: SyntaxWarning: "is" with a literal. Did you mean "=="?
Installing z3sec_control script to /usr/local/bin
Installing z3sec_install_code script to /usr/local/bin
Installing z3sec_key_extract script to /usr/local/bin
Installing z3sec_show script to /usr/local/bin
Installing z3sec_touchlink script to /usr/local/bin

Installed /usr/local/lib/python3.11/dist-packages/z3sec-0.1-py3.11.egg
Processing dependencies for z3sec==0.1
Finished processing dependencies for z3sec==0.1
```

Figure D.1: Shows the successful installation of Z3SEC.

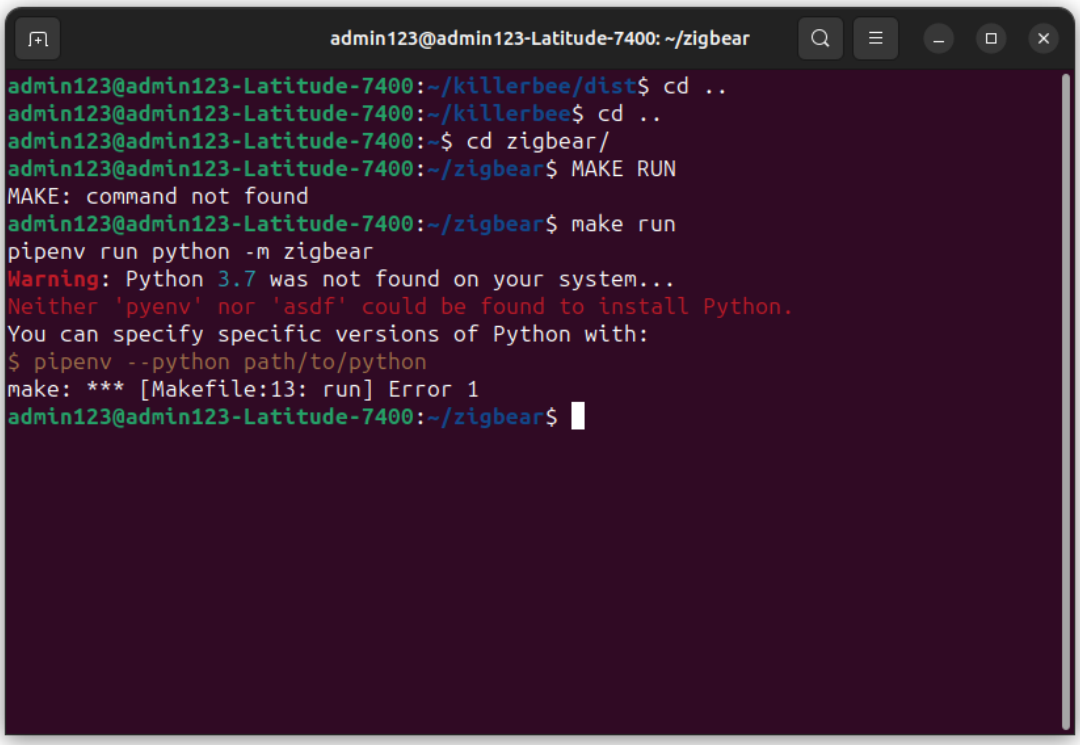

Figure D.2: Failure in the installation of Zigbear due to the lack of a compatible version of Python.

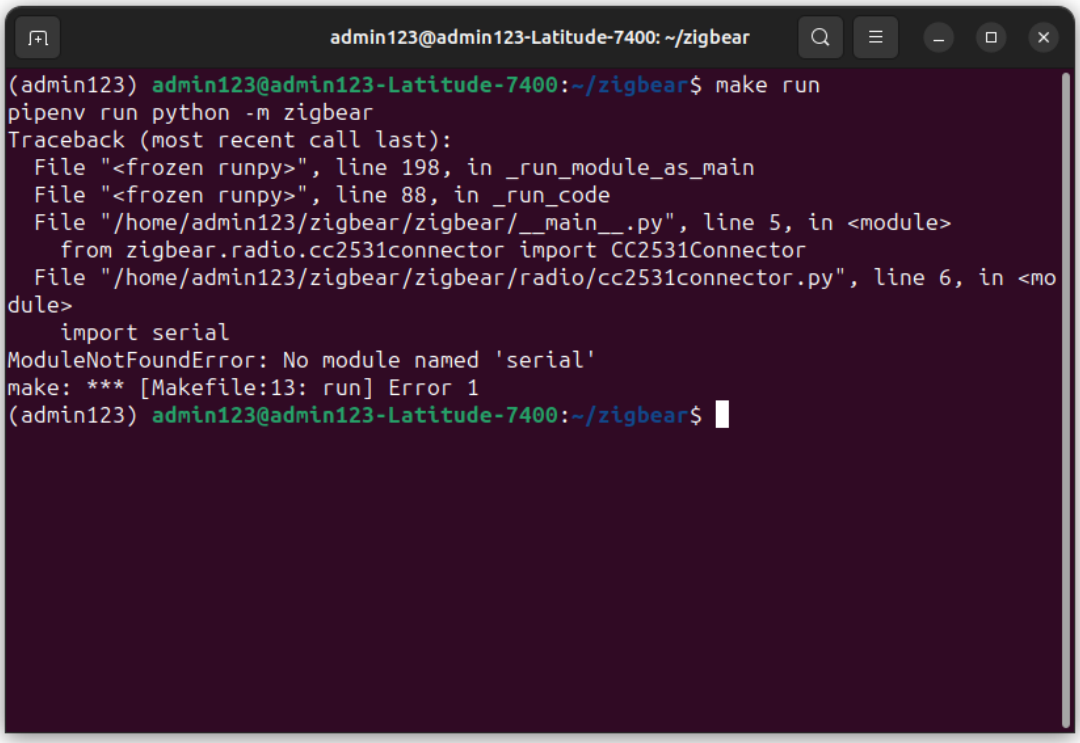

Figure D.3: Error in installing Zigbear even after downgrading Python version to 3.7.

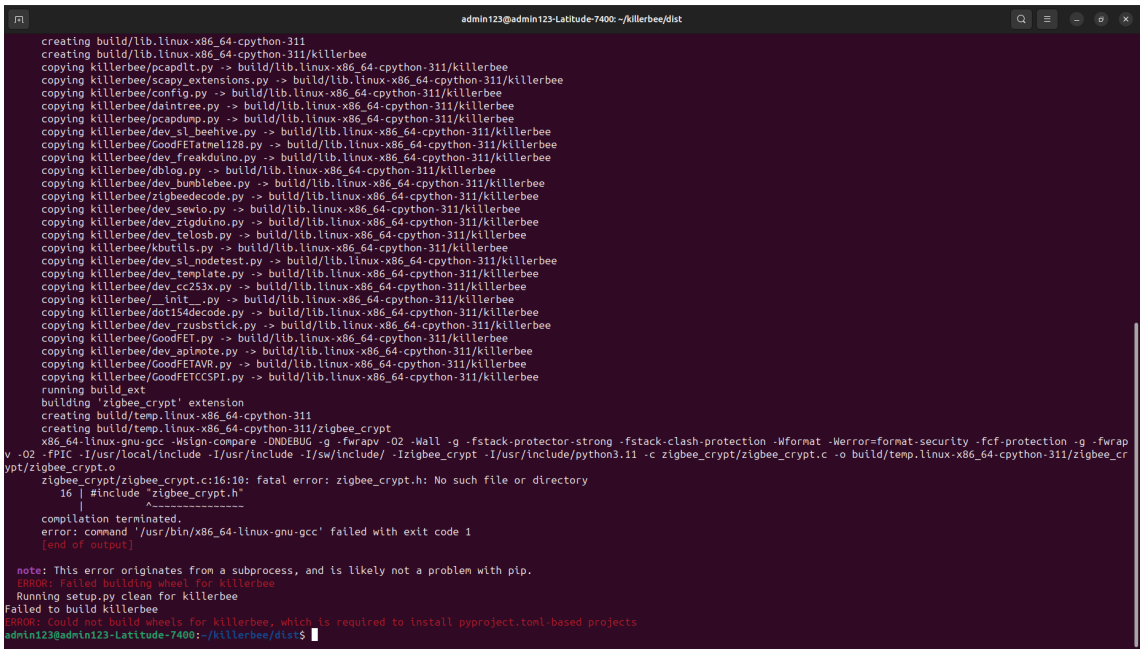

Figure D.4: Unsuccessful installation of KillerBee due to failure of building the wheel.

E Wireshark Traffic

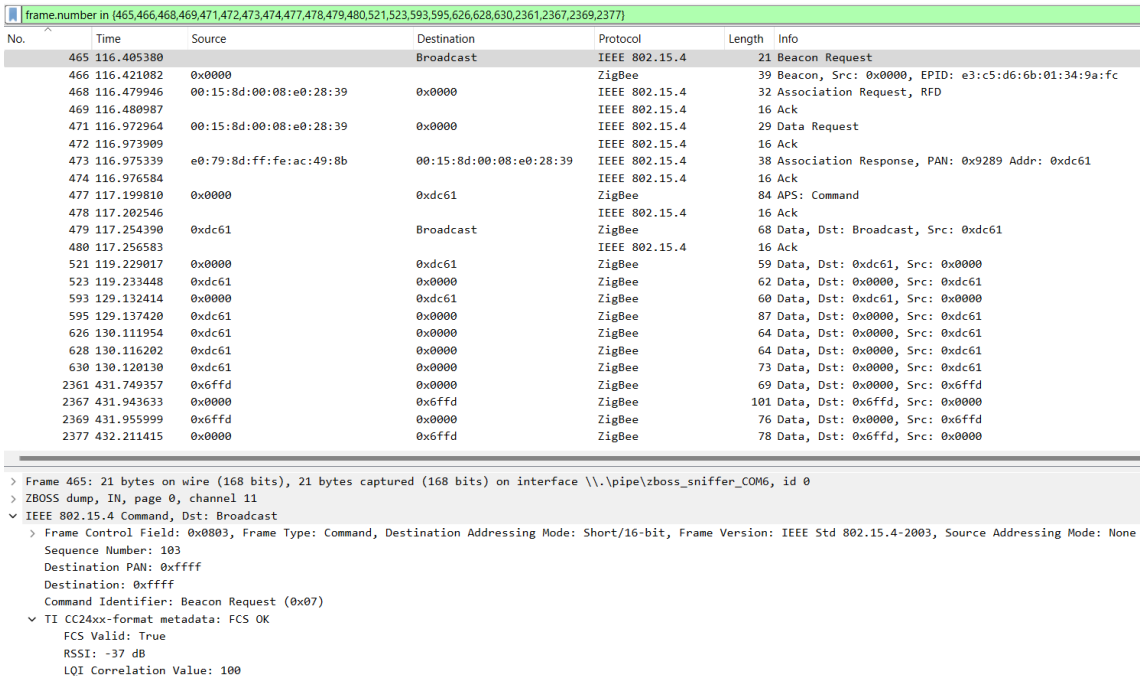

Figure E.1: Encrypted and filtered ZigBee and IEEE 802.15.4 captured packets on Wireshark.

- Github link for Wireshark captures: Wireshark captures 1 and 2