Detecting encrypted traffic activities and patterns in ZigBee network Data

Joy O. Falaye
Research and Exploratory Development Department
The Johns Hopkins University Applied Physics Laboratory
Laurel, MD
Joy.Falaye@jhuapl.edu

Jeffrey S. Chavis
Asymmetric Operations Sector
The Johns Hopkins University Applied Physics Laboratory
Laurel, MD
Jeffrey.Chavis@jhuapl.edu

Dan Simon
Asymmetric Operations Sector
The Johns Hopkins University Applied Physics Laboratory
Laurel, MD
Dan.Simon@jhuapl.edu

Khir Henderson
Asymmetric Operations Sector
The Johns Hopkins University Applied Physics Laboratory
Laurel, MD
Khhen2@morgan.edu

Kevin T. Kornegay
Morgan State University
Baltimore, MD
Kevin.Kornegay@morgan.edu

2023 IEEE Integrated STEM Education Conference (ISEC) | 979-8-3503-0001-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISEC57711.2023.10402346

Abstract—With the increase in data transmissions and network traffic over the years, there has been an increase in concerns about protecting network data and information from snooping. With this concern, encryptions are incorporated into network protocols. From wireless protocols to web and phone applications, systems that handle the going and coming of data on the network have applied different kinds of encryptions to protect the confidentiality and integrity of their data transfers. The addition of encryptions poses a new question. What will be observed from encrypted traffic data? This work in progress research delivers an in-depth overview of the ZigBee protocol and analyzes encrypted ZigBee traffic on the ZigBee network. From our analysis, we developed possible strategies for ZigBee traffic analysis. Adopting the proposed strategy makes it possible to detect encrypted traffic activities and patterns of use on the ZigBee network. To the best of our knowledge, this is the first work that tries to understand encrypted ZigBee traffic. By understanding what can be gained from encrypted traffic, this work will benefit the security and privacy of the ZigBee protocol.

Keywords: IEEE 802.15.4, IoT, ZigBee

I. Introduction

ZigBee is a wireless protocol that provides low-energy wireless communication between devices on the network. Using this low-energy data transmission, the ZigBee framework created a space for itself in the Internet of Things (IoT) field. Using the ZigBee specification, companies such as Phillips and Eero developed smart lights, motion sensors, smart thermostats, and many other smart IoT devices. The exploration of ZigBee stems from the need to understand what information users can discover from ZigBee devices. This information is recorded from the standpoint of the ZigBee devices to users and users of ZigBee devices.

Before getting into the network traffic analysis, it is essential to understand ZigBee itself. ZigBee devices run on a mesh network. Mesh networks are interconnected networks where each device or node is linked together. This link is established through routing channels. When a device sends a message in a mesh network, a path is created or followed through the node-to-node movement until it arrives at its destination. All ZigBee device nodes are interconnected through routing channels as long as the distance permits. Through mesh networks, ZigBee provides quality and reliable connections. A ZigBee network allows up to 653,356 devices [1] on the network. These devices have an estimated range limit of 50 meters between devices. In cases where devices are damaged, disconnected, or removed, mesh networks rebuild around broken node connections using self-healing algorithms. With this capability, routing connections are rebuilt or replaced with an alternate path.

II. Packet Structure

For the ZigBee wireless Framework, the protocol stack contains four layers: Physical, Data Link, Network, and Application. The structure of the ZigBee packet is shown in (Fig. 1). The layers divide into groups in the ZigBee protocol stack based on the IEEE 802.15.4 standard and the ZigBee alliance. The ZigBee protocol's Physical and Data Link layers are based on the IEEE 802.15.4 standard, whereas the ZigBee Alliance defines the ZigBee protocol's Network and Application layers. In the ZigBee Open Systems Interconnection (OSI) model, the lower layer levels encapsulate the layers above. The Physical layer in the protocol stack encapsulates the Data Link layer, and the Data Link layer encapsulates the Network layer. In the Application layer, three sublayers exist that perform different activities; these layers are all encapsulated by the Network Layer. The functions of these layers in the ZigBee Network [1] are explored next.

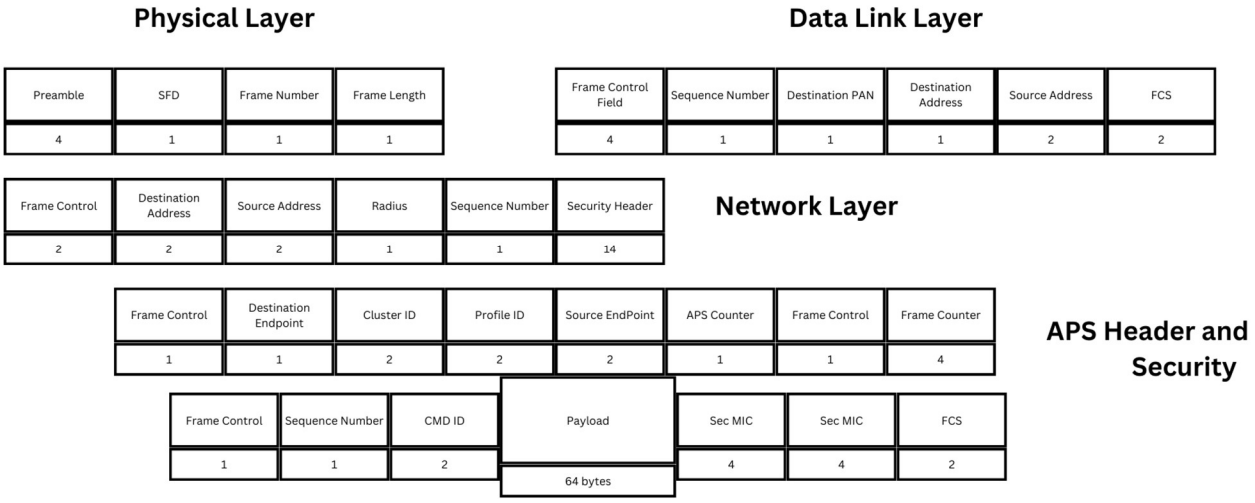

Fig. 1. ZigBee Packet Format

The Physical layer handles all tasks involving access to the ZigBee hardware:

- Initializing the hardware: Loads and powers up the device.
- Channel selection: Selects the channel on which the device wants to perform activities.
- Clear channel assessment: Assists the channel selection in finding a channel.
- Link quality estimation: Checks the strength of the device's link to other devices before initiating activity.

The Physical layer is responsible for receiving and sending the bits from one device to another while setting the parameters for the conversations between devices. These responsibilities are handled on the Physical layer.

The Data Link layer transmits data from the Network layer to the Physical layer and vice versa. The Data Link layer has two primary services: the Media Access Control (MAC) data services and the MAC management service. These services interface with the MAC Sublayer Management Entity (MLME). Through the use of the Data Link service access point called MLME-SAP, the Data Link data service enables the transmission of MAC Protocol Data Units (MPDUs) across the Physical layer. For the MAC management service, MLME-SAP allows the transport of management commands between the next higher layer and the MLME. The Data Link layer generates beacons and synchronizes devices to the signal. Four frame structures defined in the Data Link layer are Beacon, Data, Acknowledge, and MAC Command.

The Data Link layer's most significant role is facilitating communication between the Physical and Network layers. The Data Link layer's primary services translate commands from higher layers to the Physical layer and send data retrieved from the Physical layer to the Network layer.

The Network layer is responsible for the creation of the ZigBee network. To create the network, the Network layer has two primary responsibilities: pathway discovery and pathway selection. Pathway discovery occurs when the ZigBee device broadcasts a message for other devices on the network and finds all possible paths to this device. These pathways act as routing channels to send messages from device to device. After creating these routing channels, the next responsibility is to identify which channel is best to send a message to the destined device. This is the responsibility of the Pathway selection capability. The Network layer discovers routing channels and selects the best paths to the destination nodes on the network. The ZigBee network forms by creating routing channels to connect the multiple devices on the network.

The Application layer is the highest layer in the ZigBee protocol stack. The ZigBee specification separates the APL layer into three sublayers: Application Support, ZigBee Device Objects, and Application Objects.

The Application Support sublayer interfaces the network and application layers in the protocol stack. By processing the incoming and outgoing traffic between the Network layer and the Application layer, the Application Support sublayer can ensure the security of transmissions in the network/protocol stack. During this process, the application creates cryptography keys and manages them. In addition to these capabilities, the Application Support sublayer offers Key Establishment, Key Transportation, Device Updates, Device Removal, Key Requests, Key Switching, Entity Authentication, and Permissions Configuration Table services.

The ZigBee Device Objects sublayer works to secure the devices through authentication and encryption, discover services on the network, and bind the nodes to different services and applications on the network.

The Application Objects sublayer controls and manages the protocol layers in the ZigBee device. It controls the hardware in the ZigBee device and assigns a unique endpoint number that other Application Objects can use to interact with it. There can be up to 240 Application Objects in a ZigBee device. ZigBee applications must be adapted to an existing application profile

that ZigBee Alliance accepts. These profiles define message formats and protocols for Application Objects interactions. Through these application profiles, ZigBee devices can inter-operate and communicate with devices from different vendors in a given application profile.

Each layer of the ZigBee protocol stack (Fig. 2) carries an important task that helps in the regulation, initiation, and organization of activities that occur on the network.

III. What is Encrypted? What is Not?

With numerous messages and commands being sent through the network and protocol stack, it is essential to determine what is observable on the network. ZigBee is a protocol that protects the user's data through encryption. The ZigBee network uses Advanced Encryption Standard (AES) 128-bit encryption to encrypt its data and maintain the integrity and confidentiality of its network.

The ZigBee network has an open trust model [2], which means the protocol stack layers trust each other. As a result of this trust, cryptography encryptions are placed only between different devices, not between different layers. The open trust model prevents encryption between layers of the stack; however, for security purposes, a small part of the 802.15.4 frame is encrypted by the network key. This small encryption verifies the received data [2]. The encryption works with the Message Authentication Code to validate the information sender.

Although data transmission between layers in the protocol stack is relatively free, all data transmissions between the devices are encrypted on the ZigBee network. The basic security provided by the ZigBee network is data encryption using the network key. This key is based on AES 128-bit encryption, and is transported to the joining device during the authentication process. The network key is never sent over the air unencrypted, thus protecting it from a possible attacker.

IV. External vs. Internal Networks

When looking at the ZigBee network, we can use an 802.15.4 sniffer to capture the network traffic. This is over-the-air (OTA) network scanning. Another way to understand the ZigBee activity is to look at the hub application for the ZigBee devices. This hub shows the devices on the network and their current status. From OTA scanning, we can retrieve more information such as the frame length, source and destination addresses, radius, and sequence numbers. However, from looking at the hub, we get a more in-depth understanding of what is happening in the network.

This is the difference between the internal and external of the ZigBee network. Observing the ZigBee network through the hub application allows an individual to explore the network without encryption. A user can determine which of the smart lights is what color or the command sent to the other smart device. It is possible to know whether a device is deactivated and receives alerts from the motion sensor on activity. This is the internal network. On the external network, users can obtain information, but all critical data are encrypted. The information obtained is explored in the next section. Plenty of information can be retrieved, but this information does not include the core commands sent from device to device. That data is encrypted on the external network. The internal of a ZigBee network contains information on the encrypted messages sent from device to device on the network; everything else is external.

V. What can be Collected Externally and How?

In exploring what can be found in the external network, we first identified all the unencrypted variables in the network scans. We further classified the identified unencrypted data as distinguishable or non-distinguishable. An example of the WireShark output is shown in (Fig. 3). We created four tables to represent the four observable layers on the ZigBee network: Physical, Data Link, Network, and Application. We observed the information retrieved from the network sniffing and documented all unencrypted data in Tables I, II, III, and IV.

In the Physical layer of the ZigBee Protocol stack, the Encapsulation Type, Time, Frame Number, Frame Length, and Capture Length are unencrypted (Table I).

In the Data Link layer of the ZigBee Protocol stack, the Frame Control Field, Frame Type (Data), Frame Version, Sequence Number, Destination PAN, Destination, Source, and Extended Source are unencrypted (Table II).

In the Network layer of the ZigBee Protocol stack, the Frame Control Field, the Frame Type, Destination, Source, Radius, Sequence Number, and Extended Source are unencrypted. In the ZigBee Security Header, the Security Control Field, Key ID, Frame Counter, Extended Source, Key Sequence Number, and Message Integrity Code are unencrypted (Table III).

In the Application layer of the ZigBee Protocol stack, the Frame Control Field, Destination Endpoint, Cluster, Profile, Source Endpoint, and Counter are unencrypted. (Table IV).

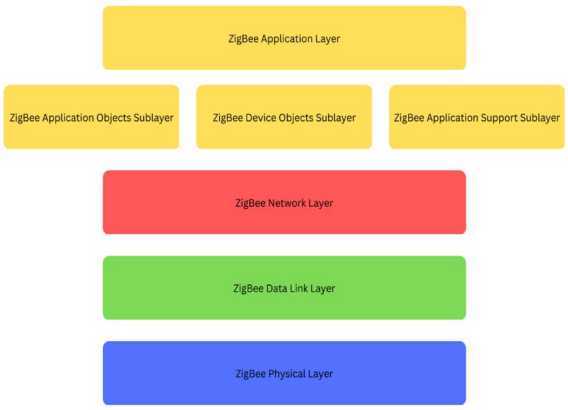

Fig. 2. ZigBee Protocol Stack

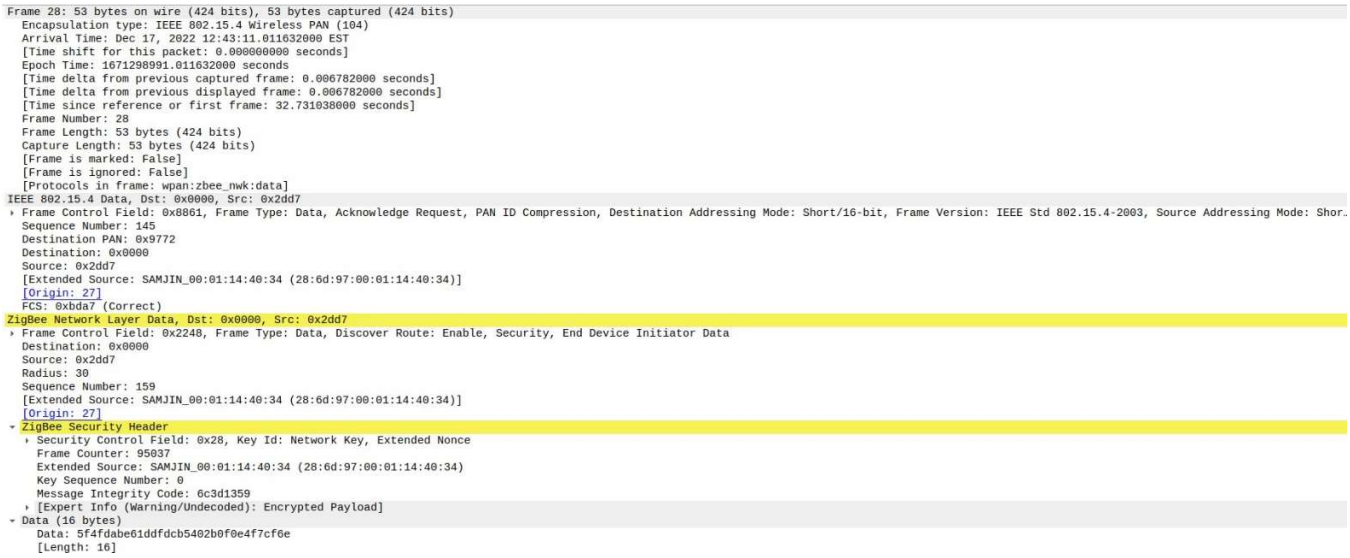

Fig. 3. Captured ZigBee Packet

TABLE I. UNENCRYPTED DATA IN PHYSICAL LAYER

Frame (IEEE 802.15)
Encapsulation Type (IEEE 802.15.4 Wireless PAN)
Time (Arrival and Epoch)
Frame Number
Frame Length
Capture Length

TABLE II. UNENCRYPTED DATA IN THE DATA LINK LAYER

IEEE 802.15.4
Frame Control Field
Frame Type (Data)
Frame Version (IEEE Std 802.15.4-2003)
Sequence Number
Destination PAN
Destination
Source
Extended Source (PhilipsL01 : 05 : 13 : f 9 : 4d)

TABLE III. UNENCRYPTED DATA IN NETWORK LAYER AND SECURITY HEADER

ZigBee Network Layer
Frame Control Field
Frame Type (Data)
Destination
Source
Radius
Sequence Number
Extended Source (OUI)
ZigBee Security Header
Security Control Field
Key ID
Frame Counter
Extended Source (OUI)
Key Sequence Number
Message Integrity Code

TABLE IV. UNENCRYPTED DATA IN APPLICATION LAYER

ZigBee Application Layer
Frame Control Field
Destination Endpoint
Cluster
Profile ID
Source Endpoint
Counter

VI. DISTINGUISHABLE INFORMATION

Distinguishable information is identified as data that is found to be clearly different or recognized compared to other data points. For the purpose of this research, it represents data points in the packet structure that show a visible change from packet to packet. Distinguishable data plays an important role in this research; it acts as possible features to be extracted for data classification.

The Packet Frame carries four distinguishable variables (Table V):

- Time (Arrival and Epoch) – This represents the time of arrival of the packet to its destination. This is a distinguishing feature because not every packet arrives at the same time. With this data point, it is possible to relate certain actions taken by ZigBee devices with the packet sent.
- Frame Number – This represents the order in which packet data is collected by WireShark. This number, along with other information like time, can help us understand the timeline at which events occur on the network.
- Frame Length – This shows the total length of the frame sent. Different lengths for the frame can identify different messages being sent.

- Capture Length – This shows the total length of the frame length captured. Different lengths for the frame can identify different messages being sent.

TABLE V. DISTINGUISHABLE DATA IN PHYSICAL LAYER

Packet Frame (IEEE 802.15) Distinguishable
Time (Arrival and Epoch)
Frame Number
Frame Length
Capture Length

TABLE VI. DISTINGUISHABLE DATA IN DATA LINK LAYER

IEEE 802.15.4 Distinguishable
Frame Type
Sequence Number
Destination
Source
Extended Source (OUI)

TABLE VII. DISTINGUISHABLE DATA IN NETWORK LAYER

ZigBee Network Layer Distinguishable
Frame Type
Destination
Source
Radius
Sequence Number
Extended Source (OUI)
ZigBee Security Header
Frame Counter
Message Integrity Code

TABLE VIII. DISTINGUISHABLE DATA IN APPLICATION LAYER

ZigBee Application Layer Distinguishable
Destination Endpoint
Profile ID
Source Endpoint
Counter

IEEE 802.15.4 carries five distinguishable variables (Table VI):

- Frame Type – This identifies the data format of the data being sent by the packet.
- Sequence Number – This represents the Beacon Sequence number or Sequence Identifier for the frame.
- Destination – This is the address to which the packet is being sent.
- Source – This is the address from which the packet is being sent.
- Extended Source – This represents the MAC address of the sender of the packet. This address, if documented in the Organizationally Unique Identifier (OUI), can provide the identity of a manufacturer or brand of the sender.

The ZigBee Network layer carries eight distinguishable variables (Table VII):

- Frame Types – Explained previously.
- Destination – Explained previously.
- Source – Explained previously.
- Radius – This represents the number of hops remaining for a range-limited broadcast packet.
- Sequence Number – This is a number that is incremented to allow devices to identify which instance of the network key has been used to secure the packet data.
- Extended Source – Explained previously.
- Frame Counter – This is incremented to allow devices to identify which instance of the network key has been used to secure the packet data.
- Message Integrity Code – This is used to authenticate the message by ensuring it has not been modified.

The ZigBee Application layer carries eight distinguishable variables (Table VIII):

- Destination Endpoint – This is an integer between 0 and 240. Applications register with this for verification when entering the Application Layer.
- Source Endpoint – This is an integer between 0 and 240. Applications register with this for verification when entering the Application Layer.
- Profile – This is an identifier for the application running on the network.
- Counter – This is a value that increments with each command in the Application layer.

VII. WHAT CAN BE COLLECTED INTERNALLY AND HOW?

In determining what can be found in the internal network, we identify all data values shown to be encrypted during the network scan. This turned out to be the network key used to encrypt the messages and the data portion that carries the command or reply being sent from one device to the other (Table IX).

The experimental setup of our experiment is shown in (Fig. 4). For this experiment, we connected 16 Texas Instruments CC2531emk USB dongles through 2 USBGear multi-port devices. Using KillerBee, an open source ZigBee pen testing system, sniffing is run through the Killerbee openear code. Each dongle sniffs one of the 16 ZigBee channels. We are collecting OTA traffic data through ZigBee packet sniffers. The packet information gathered is separated into four sections that cover the Frame, 802.15.4 Specification, ZigBee Network Layer and ZigBee Application Layer. Each section of the packet represents the different levels of the protocol stack. Information on what is in each level of the packet was previously discussed. In addition to the ZigBee protocol, the CC2531emk USB dongles also collect other protocols in the 802.15.4 specification. Thread, 6LoWPAN, and Lightweight Mesh are protocols that were observed in our packet capture. Previously,

we worked with the open-source pyCCSniffer. This code allowed for sniffing, but only worked with one channel per run. Another issue was the confusion created by the sniffed data. It was not clear where on the protocol stack this data came from. This new sniffing works using WireShark. This gives a more formatted structured output, allowing a clear understanding of the packet structure of ZigBee devices.

TABLE IX. ENCRYPTED DATA IN ZIGBEE NETWORK PROTOCOL EXPERIMENTAL SETUP

ZigBee Network Layer
Network Key
Data

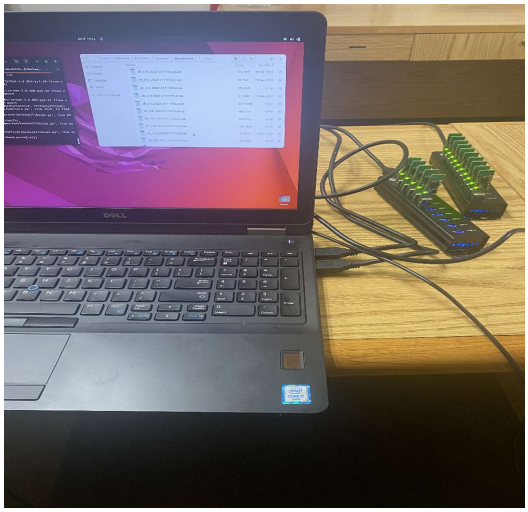

Fig. 4. ZigBee Experimental Setup

VIII. DISCOVERY: WHAT HAS BEEN WITH DONE WITH SIMILAR INFORMATION?

Network traffic analysis is the key to understanding encrypted traffic data. In essence, network traffic analysis is finding patterns and correlations on the network. When analyzing a network, it is crucial to understand the goal of the analysis. From the patterns found, we can use this data to classify activities, discover a pattern of use, and obtain system information. Ly Vu et al. [3] used network analysis to classify network traffic. Their research used payloads and flow-based methods to develop a time-series–based network analysis to understand the network traffic behavior. Using this time series analysis, the authors used the patterns found in these time series data points to develop a time series feature extractor to create a classification model that can classify the network data. Time series analysis is also essential for finding the pattern of use. Using time series data makes it possible to identify a schedule on the network for certain activities.

Chengshang Hou et al. [4] analyzed encrypted WeChat data transmissions. Their study observed MMTLS-encrypted traffic to understand retrievable data from encrypted WeChat traffic. Their analysis discovered patterns in the network traffic correlated to specific activities on the application. The authors using machine-learning, classified actions on the network with their corresponding actions on the messaging application. WeChat was one of many instant messaging (IM) applications to be researched in such a way. To understand and analyze encrypted networks, Asmara Afzal et al. [5] explored the signal protocol and IM application. The authors' research outlined the five steps to analyzing and classifying encrypted network traffic: performing activities, capturing traffic, analyzing traffic patterns, verifying traffic analysis, and compiling results. Using these steps, the authors could classify network traffic into activities performed on the application.

This encrypted traffic analysis does not just exist for IM applications. Jonathan Muehlstein et al. [6] examined encrypted web data by analyzing the HTTPS protocol. Unlike classification, this research aimed to obtain system information from the analysis result. By exploiting the traffic patterns found in the network analysis of the encrypted data, the authors were able to build a model that identifies the operating system, the browser, and the applications running on that browser.

The encrypted network analysis on IM and web applications shows the practicality of doing these things with ZigBee. When reading these papers, it was also important to find examples of analysis done on wireless low-area networks like ZigBee or similar protocols. While no papers were discovered for ZigBee, research by Jeffrey Chavis et al. [7] did something similar for Wi-Fi. Their research used Wi-Fi traffic data to create a classification model that identified the manufacturer and type of device used on the network.

Encrypted network traffic protects payload data from direct intrusion, but traffic analysis and research can still reveal many things.

IX. ANALYSIS METHODS

After establishing a baseline of the ZigBee packet structure, we reviewed the literature to discover potential analysis methods and other use cases. We identified several analysis methods, including developing a pattern of device usage, classification of devices by type, and classification of activity on the network. The first method is heavily researched and developed. This method involves using encrypted information to develop a timeline of device usage. To create a timeline, understanding the devices on the network is crucial. Using time series analysis and the source and destination addresses on the network makes it possible to identify a pattern of use for devices on the network. In applying this to our ZigBee packet structure, it is possible to develop a pattern of usage that allows us to create a network schedule and predict which devices are used at certain times of the day.

Similar to the pattern of usage method, the classification method is also heavily researched and developed. In our analysis, we identified two types of classification: classification of devices by type and classification of activity on the network. Classification of activity correlates network activity to ZigBee activity, making it possible to classify specific actions with their corresponding network imprint. Similarly, classification of devices by type correlates the types of activities performed by devices to their possible device types. These two types of classification work together to create a clearer picture of the network. To develop the activity classification for the ZigBee network, we follow the classification steps provided by Asmara

Afzal et al. [5]. We can then classify the devices on the network using the activity classification output.

Potential analysis methods show that current methods can apply to the ZigBee protocol. These methods cannot only apply, but can work together to form a more comprehensive understanding of the ZigBee network. Combining all these methods makes it possible to create a timeline of the ZigBee network that shows how many devices are on the network, their activities, and the types of devices.

X. Approach

After performing our literature review and comparing packets, we decided two methods are applicable to the ZigBee packet schema: classification and pattern of use. Because our goal is to discover everything we can using the packets, we discuss all approaches mentioned.

Our first approach is classification of ZigBee packets as a method of discovery. Our previous research showed similarities in packet structure. The classification method essentially gathers data, labels the information, and extracts features that are distinguishable. Features such as time, radius, frame counter, and size are extracted in the development of the classification model of this protocol.

For the ZigBee packet classification, the first step is to gather network data. This data draws from activities occurring on the network. The data drawn corresponds to each possible ZigBee activity on the network. From turning on the light to changing light colors, capturing motion-sensed data, and changing the temperature, these activities are captured and labeled on the network. The next step is network analysis. From each labeled activity, distinguishable features are extracted. These features include length, frame type, sequence number, and radius. Using these features and a classification model of choice, a network activity classifier can be created. For the final step, the model created is tested for accuracy and other parameters to verify the analysis results.

At the end of this approach, we expected to classify activities such as “change bulb color,” “motion detected,” and “change temperature.” The result of this classification will be the awareness of network activities present on the network.

ZigBee device classification works as an extension of the activity classification. Using the output created by the activity classifier, device classification uses activities to identify devices. This approach inputs address information and corresponding activity information to determine device types. When a device location is attached to a “change bulb color” command, the device would be labeled a smart lightbulb.

At the end of this approach, we expected to classify activities such as “light device,” “motion sensor,” and “smart thermostat.” This classification will result in awareness of network devices present on the network. Our second approach is to identify the pattern of use of ZigBee packets as a discovery method. In previous research, time series analysis was used to identify network patterns and develop classification models. The pattern of use is found by using time series analysis to identify a schedule of use for device on the network. A pattern of use can be found when a device address is observed to continuously send data, retrieve data, or engage in conversations on the network over a specific timeframe. The analysis works to find network activity that correlates certain times of the day to specific activities on the network. To find use patterns, data collection and data querying are used to sift through network data and find clues. When the network activity is selected, the goal is to measure whether this scenario is repeated multiple times and, if so, what is the basis. That basis is the pattern of use.

At the end of this approach, we expected to identify network patterns of use for the devices on the network. This approach will result in a higher probability of predicting network activities based on previous patterns found.

This approach can be built on top of the classification methods previously discussed. With the ZigBee network better explained through classification, it is easier to identify activities and recognize use patterns occurring.

XI. Conclusions

In this paper, we identified the ZigBee and its use cases, capabilities, and structure. We also proposed various strategies for ZigBee traffic analysis. Without performing penetration testing on the ZigBee network, we observed possible behavior from encrypted network traffic of the ZigBee network protocol. The ZigBee network traffic does not provide much information on device type, network activity, and usage patterns, but this information can be extracted through encrypted network analysis. The information gained through such analysis benefits the development of a secure ZigBee architecture.

XII. Future Work

In the future, we will implement the strategies proposed in this paper. Our current strategies identify both use patterns and classification network activities and devices. In subsequent work, we will experiment with our developed strategies to test their effect. Our work will focus on developing encrypted network analysis tools and, when complete, will help identify more optimal and accurate methods for the analysis.

XIII. References

[1] C. M. Ramya, M. Shanmugaraj, and R. Prabakaran, “Study on zigbee technology,” vol. 6, 2011, pp. 297–301.

[2] X. Fan, F. Susan, W. Long, and S. Li, “Security analysis of zigbee,” 2017.

[3] L. Vu, H. V. Thuy, Q. U. Nguyen, T. N. Ngoc, D. N. Nguyen, D. T. Hoang, and E. Dutkiewicz, “Time series analysis for encrypted traffic classification: A deep learning approach,” 2018.

[4] C. Hou, J. Shi, C. Kang, Z. Cao, and X. Gang, “Classifying user activities in the encrypted wechat traffic,” 2018.

[5] A. Afzal, M. Hussain, S. Saleem, M. K. Shahzad, A. T. Ho, and K. H. Jung, “Encrypted network traffic analysis of secure instant messaging application: A case study of signal messenger app,” Applied Sciences (Switzerland), vol. 11, 2021.

[6] J. Muehlstein, Y. Zion, M. Bahumi, I. Kirshenboim, R. Dubin, A. Dvir, and O. Pele, “Analyzing https encrypted traffic to identify user’s operating system, browser and application,” vol. 2017-January, 2017.

[7] J. S. Chavis, A. Buczak, A. Rubin, and L. A. Watkins, “Connected home automated security monitor (chasm): Protecting IoT through application of machine learning,” 2020.