REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Electrical and Computer Engineering

SURVEY OF CYBER SECURITY ISSUES IN INTERNET OF THINGS (IOT)

Abdulraheem Ali Mohsin AL-FATLAWI

Master's Thesis

Supervisor

Asst. Prof. Dr. Ayça KURNAZ TÜRKBEN

Istanbul, 2023

SURVEY OF CYBER SECURITY ISSUES IN INTERNET OF THINGS (IOT)

Abdulraheem Ali Mohsin AL-FATLAWI

Electrical and Computer Engineering

Master's Thesis

ALTINBAŞ UNIVERSITY

2023

The thesis titled SURVEY OF CYBER SECURITY ISSUES IN INTERNET OF THINGS (IOT) prepared by ABDULRAHEEM ALI MOHSIN AL-FATLAWI and submitted on 03/05/2023 has been **accepted by majority of votes** for the degree of Master of Science in Electrical and Computer Engineering.

Asst.Prof. Dr. Ayça Kurnaz TÜRKBEN

Supervisor

Thesis Defense Committee Members:

Asst.Prof. Dr. Ayça Kurnaz TÜRKBEN	Department of Software Engineering, Altınbaş University	_____
Asst. Prof. Dr. Abdullahi Abdu IBRAHIM	Department of Computer Engineering, Altınbaş University	_____
Asst. Prof. Dr. Zeynep ALTAN	Department of Software Engineering, Beykent University	_____

I hereby declare that this thesis meets all format and submission requirements of a master's thesis.

Submission date of the thesis to Institute of Graduate Studies: ___/___/__

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Abdulraheem AL-FATLAWI

Signature

ABSTRACT

SURVEY OF CYBER SECURITY ISSUES IN INTERNET OF THINGS (IOT)

AL-FATLAWI, Abdulraheem

M.Sc., Electrical and Computer Engineering, Altınbaş University,

Supervisor: Asst. Prof. Dr. Ayça Kurnaz TÜRKBEN

Date: April / 2023

Pages: 52

The Internet of Things (IoT) has revolutionized the way we live, work, and communicate. With billions of connected devices communicating with each other. However, the widespread use of IoT devices has also created new challenges in the area of cyber security. Cybersecurity threats pose a serious risk to the IoT, which can compromise the privacy, safety, and security of users, organizations, and governments. The vast amount of sensitive data transmitted over IoT networks can make them a target for attacks and cyber criminals. Additionally, IoT devices often lack the security measures required to prevent unauthorized access, data theft, and other cyber-attacks. As a result, the rapid growth of IoT devices has created an urgent need for organizations to implement robust cyber security measures that can prevent and mitigate these threats. This requires an integrated and coordinated approach involving government, industry, and academia to develop and implement effective cybersecurity solutions and standards that can protect the IoT and its users. This thesis examined the architecture supporting Internet of Things devices, data aggregation facilities, and communication routes. It then described device, network, and application assaults against a system. Trusted Execution Environment and Trusted Platform Module, the two most common IoT solutions, will be discussed in the second part. It described each approach's main characteristics, working principles, and drawbacks. This shows that solutions are successful but need to be modified over time. The latter section used artificial neural networks to forecast DoS attacks. This thesis illustrates that IoT security may be improved simply. The norm should ultimately standardize Internet of Things device safeguards.

Keywords: IOT, Cybersecurity, Attacks, Threats, Neural Network, DoS Detection.

TABLE OF CONTENTS

Pages

TABLE OF FIGURES

Pages

ABBREVIATIONS

RNN	:	Recurrent Neural Network
DBN	:	Deep Belief Network
DL	:	Deep Learning
IoT	:	Internet of Things
NFC	:	Near-Field Communication
WSN	:	Wireless Sensor Networks

1. INTRODUCTION

In recent years, Internet of Things (IoT) has developed into a more pervasive component of our everyday life. In today's world, Internet of Things (IoT) gadgets that are smart and linked to a network are almost ubiquitous. We have them in our houses in the shape of home security systems or technologies that make our homes more intelligent. As intelligent watches, we wear them constantly on our wrists and take them with us wherever. They also have a broad variety of uses in the industrial sector.

Despite the fact that these devices are intended to collect information that is both private and sensitive to security, and despite the sheer number of these devices that are currently in use (more than 30 billion units are predicted to be in use by 2025 [1]), they do not have a good reputation for keeping data safe from potential attackers. On the contrary, Internet of Things is often stated combined with a lack of standards when security is being discussed. The wide variety of possible configurations of the devices is the primary source of the challenges inherent in standardizing. This means that the issue of security is generally left up to the creator or supplier of the device itself, which means that it is often treated as more of an afterthought rather than a key priority. The number of people using the Internet continues to rise, and in April of 2022, there were 5 billion of them. Almost two-thirds of Earth's current population may be accounted for here (Statista, 2022f). Throughout, the proportion of the population with access to the internet varies greatly. In Northern Europe, for example, that number may approach 100%. [2]. This is also true with regards to Switzerland, the UK, and South Korea [3]. Internet usage has permeated daily life in every developed country and most of the rest of the world. Since the number of people with internet access and the speed at which information can be sent has increased, more and more products and services that rely on the internet have entered the marketplace in recent years [4]. Further technological advances in areas such as wireless networks, low-cost high-performance hardware, machine learning, and Artificial Intelligence (AI) will push the spread of these fields in the coming years [5]. In concert with these technical developments, changes are occurring in the consumer landscape. The increasing demographic heterogeneity of modern societies is giving rise to more diversified needs and preferences. Different individuals may have various needs in terms of simplicity, durability, or even specialized care, all of which are examples of how expectations show themselves [6]. As it's growing harder to successfully sell to a wide variety of target groups, this presents a number of challenges for marketers [6]. In order

to meet the ever-expanding variety of customer demands, firms must make ever-finer adjustments than in the past [7].

Technological progress and the increasing normalization of such developments in our daily lives have given businesses the opportunity to alter the ways in which they offer their products and services [8]. In particular, IoT enables connections to be made between the digital and physical realms. When more and more items become equipped with networked sensors capable of communicating with one another, new opportunities arise not just for customers but also for objects themselves. Business opportunities arise as a result of the interconnectedness and communication of the things. For instance, it could be easier to gauge customer tastes, leading to more customized products and services. By considering various types of data, they improve the overall customer experience and provide value for the client [9]. There are several arguments that suggest IoT might drastically alter the way consumers engage with businesses [10]. Thus, there may be substantial opportunity for organizations to enhance the efficiency of their marketing thanks to internet-connected gadgets and, more especially, the data that these devices give. But user consent is required before Internet of Things applications and the data they generate may be used to their fullest potential [9]. As this is the case, this thesis will focus on how well Internet of Things technology is received. In this case, we'll use the wristwatch as a model. Other marketing uses for the Internet of Things have been identified and explored as a result of these and other studies.

2. INTERNET OF THINGS (IOT)

2.1 BACKGROUND

The history of the phrase "Internet of Things" as well as its explanation are provided here with the purpose of fostering a deeper comprehension of this topic. It is generally agreed that Kevin Ashton came up with the concept of the Internet of Things first. In 1999, he gave a presentation at Procter & Gamble (P&G), where he was the first person to use the phrase in connection with connecting Radiofrequency-Identification (RFID) and the supply chain of the corporation [10]. Hence, radio frequency identification (RFID) technology served as the foundation for the Internet of Things. By the use of RFID transponders, automated and contactless identification was made possible with the assistance of RFID technology [11]. Historically, all of the data was gathered only by people, who are subject to the constraints of time and accuracy. The Internet of Things makes it possible to gather data in a precise and automated manner, which in turn helps to significantly cut down on waste, loss, and costs (Ashton, 2009). Nowadays, "networked connectivity of ordinary items" is what people mean when they talk about the Internet of Things (also known as the Internet of Objects) [13]. These objects may either represent actual physical items that are capable of computation and communication, or they might represent intelligent components or intelligent gadgets. Under the context of the Internet of things, these devices come together to create a network that is able to gather data via the use of sensors, interact with one another, and analyze data. Another common function of Internet of Things devices is that they may store the data that is acquired. Not only are the gadgets able to talk with one another, but they also have the ability to interact with people. The Internet of Things is distinguished by the fact that it makes data transfer possible, which in turn makes it possible for computation to be carried out in a different place or by a different device [12].

Hsu and Lin (2016) provided the following description of the Internet of Things: "Whereas in the conventional internet, people were connected to each other in order to exchange information, in the IoT, machines and objects equipped with sensors are connected to each other in order for them to communicate autonomously over the internet".

2.2 THE BEGINNINGS OF THE INTERNET OF THINGS

The evolution of the Internet of Things (IoT) may be traced back to a variety of technical breakthroughs that occurred throughout history [13]. A significant number of these developments have had an effect on our day-to-day lives and, in particular, have led to the development and proliferation of the internet. Only the most significant historical events and publications that are directly relevant to the Internet of Things will be discussed in the following timeline. This is done to prevent becoming too technical or straying outside the purview of this thesis.

Figure 2.1: The history of IoT [5].

In recent years, Internet of Things (IoT) devices have been able to be employed in an increasing number of application areas. The 2.4th section goes into detail on them. In addition, the number of devices that are connected to the internet of things (IoT) has seen a

significant rise in the last few years. This new information may be found in section 2.6, which goes into further depth.

2.3 COMPONENTS OF IOT-SYSTEMS

The (IoT) technology roughly consists of four components: sensors/devices, connectivity, data processing and user interface. They are presented and described briefly in the following.

2.3.1 SENSORS AND DEVICES

The sensors and computing devices that can communicate with the physical environment and make it possible to capture and analyze data are an essential part of Internet of Things (IoT) systems [15]. Hardware components known as sensors are responsible for registering states in the physical environment and converting those states into information that can be read by machines and sent to other things [16]. Optical sensors, temperature sensors, proximity sensors, position sensors, occupancy sensors, motion sensors, velocity sensors, and occupancy sensors are some examples of different types of sensors [17]. RFID, near-field communication (NFC), and wireless sensor networks (WSN) are examples of applications that fall into the simpler category. Devices are pieces of hardware that typically have a CPU and storage space [21]. They are able to analyze the information that is received from sensors, and in certain cases they even have sensors built in [20]. Devices connected to the Internet of Things may reside in a stationary position, be transported, or even be attached to the body [22].

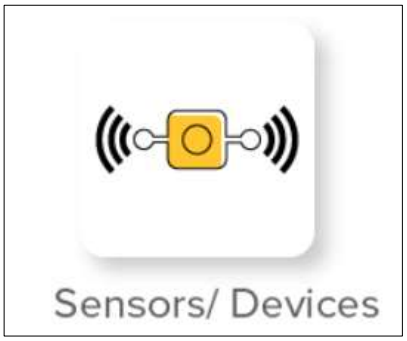

Figure 2.1: Sensors and Devices.

2.3.2 CONNECTIVITY

The Internet Protocol address, or IP address, is the fundamental component of network communication. It is a numeric number that enables one device to communicate with another device. Access to a local area network that is managed by a router may be restricted using private IP addresses. Cords, cables, and wireless technologies such as satellites, cellular networks, Wi-Fi, and Bluetooth may all be used to link the many things in the world.

The rapidly developing fifth generation (5G) network is a significant factor in Internet of Things (IoT) connection since it enhances, among other things, capacity, speed, and dependability (I-Scoop, 2018). The Internet of Things (IoT) in conjunction with fifth generation (5G) wireless networking is viewed as a potential game-changer for the next generation [20]. The presently ubiquitous 3G and 4G networks are not yet completely suitable for Internet of Things applications. This may lead one to the conclusion that, because to restricted connection, the Internet of Things is only using a small portion of its potential at the present time.

Figure 2.2: Connectivity.

2.3.3 DATA PROCESSING

Several technologies are used to store and handle the data gathered by IoT devices. At this stage, the data collected by the sensors is transformed into information that is accessible to and comprehensible by people. This information may afterwards be put to use in the actual world to conduct the right actions, either manually or automatically, depending on the situation. In particular, long-term data storage and processing is now readily achievable because to the declining cost of computer resources and cloud computing. In the future years, it is anticipated that there will be more gains in processing power in accordance with Moore's law.

Figure 2.3: Data Processing.

2.3.4 USER INTERFACE

The term "user interface" refers to the control surface that may be found on a computing device. This surface provides a graphical representation of the data collected by the sensors and, in most cases, provides additional choices for using the data or services [26]. It serves as the primary point of contact with human beings and enables communication with such individuals. In the context of the internet of things, they are often apps running on a smartphone or a wearable.

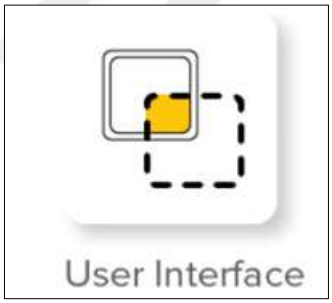

Figure 2.4: User Interface.

2.4 APPLICATION AREAS OF IOT

This chapter first provides an overview of the application areas of IoT as a whole, before presenting the most important areas from the consumer's point of view.

2.4.1 OVERVIEW

The following overview of the emerging IoT applications briefly presents the most important fields:

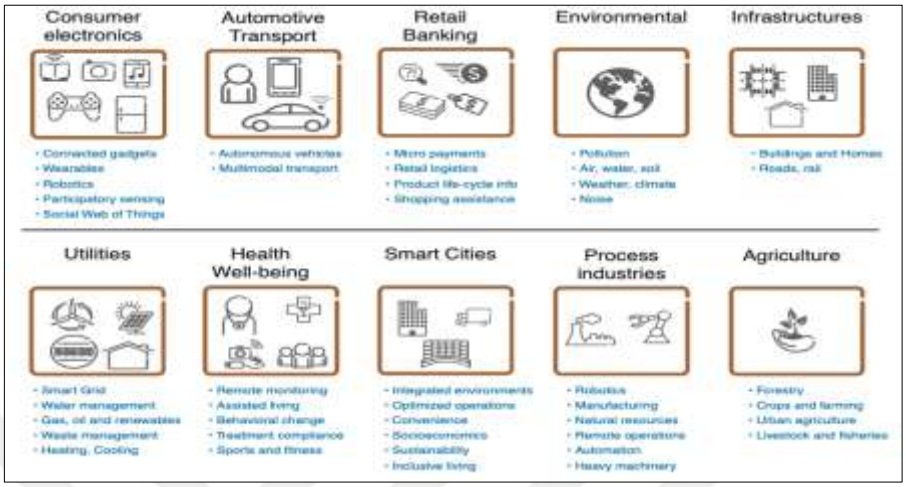

Figure 2.5: Overview IoT Applications.

The overview gives the impression that the possible applications of IoT are quite varied and might have an effect on almost every facet of our everyday life. There are several aspects that have a direct bearing on end users and have the potential to have a significant influence on them: Electronics for consumers, automobiles and modes of mobility, public utilities, and health and well-being Some sectors, including retail and finance, as well as the process industries and agriculture, provide particularly promising opportunities for businesses. But governments also have opportunities in fields such as environmental protection, city infrastructure, and smart city development. As the citizen or end customer might also be impacted by the activity of enterprises and governments, there is little doubt that a precise delimitation of the application areas is not achievable. For this reason, and in order to remain focused on the major research subject of the study, the following discussion will be limited to the most significant application domains of IoT for end users. In contrast to the graphic that provided an overview of the topic, this one is broken down into subtopics such as "smart home," "entertainment," "tracking and monitoring," and "wearables." These subtopics provide a more visual description of the different aspects of life and are frequently used in the research. Yet, it is important to point out that a distinct demarcation between these regions is not always feasible, and that there is the possibility of overlaps occurring (especially with tracking & monitoring and entertainment).

2.4.2 SMART HOME

The phrase "smart house" refers to any residence that has electronics and appliances that are networked together [27]. House automation encompasses a wide range of facets, including but not limited to the following: heating, the locking system, the doorbell, and lighting. Home appliances may range in size from huge equipment like washing machines and refrigerators to little devices like toothbrushes and vacuum cleaners. Washing machines and refrigerators are two examples of larger home appliances. In some circumstances, there are application areas that cross with those of the entertainment sector, such as voice assistance systems. Sensors are a common component of the "smart home" technologies that are designed to allow automatic control of various home appliances [28]. An application on the user's smartphone is often used in order to operate or configure the devices. The Internet of Things (IoT) in smart homes provides customers with increased convenience, time savings, security, and the possibility for cost savings via optimal management of their expenditures on power and heating.

2.4.3 Entertainment

The phrase "Smart Entertainment" refers to anything that can connect to the internet, including networked TVs, gaming consoles, toys, and voice assistance devices. In order to allow voice control or communication with other people, for example through gaming consoles, such gadgets often include microphones as an integral component. The entertainment industry also includes popular streaming services like Netflix and Spotify, which, owing to an Internet connection, provide unrestricted access to various forms of media with the assistance of Internet of Things (IoT) devices like smart TVs, smart speakers, and smartphones [29].

2.4.4 TRACING AND KEEPING AN EYE ON THINGS

The consumer sector focuses mostly on location tracking and health monitoring as its primary forms of tracking and monitoring. There is some duplication of effort between the application areas of wearables and health monitoring. In some instances, health monitoring requires tight collaboration between medical professionals and monitoring systems. There are also numerous basic measuring devices available, such as smart scales and gadgets that

measure blood pressure or pulse. These may be found in many stores. Tracking a location may be utilized not just to find goods, but also family members or pets. Real-time analysis of the data gathered here makes it possible to issue warnings or carry out actions [31].

2.4.5 WEARABLES

Wearables are Internet of Things devices that are worn on the bodies of users. There is a possibility that this application area overlaps with the fields of surveillance and monitoring as well as the entertainment industry. Such examples of wearable technology are fitness bracelets and smartwatches. Smartwatches and fitness trackers both show the time, but they also feature sensors that may record other types of data, such as movement or medical readings. Other examples include smart eyewear and clothing with embedded electronics. Bluetooth is the usual method by which these gadgets establish a connection with the user's smartphone. Moreover, certain smartwatches have the capacity to establish a direct connection to the mobile network by means of specialized SIM cards [32].

2.4.6 SMARTWATCHES

Smartwatches are wearable, as just described. Since this study examines technology acceptance using the smartwatch as an example, this IoT device will be presented in more detail now. In the academic literature, there is not one single definition of smartwatches that is universally acknowledged. According to the definition provided by Dehghani et al. (2018), a smartwatch is "a versatile wrist-worn gadget that allows quick and simple access to information and apps through a wireless or Bluetooth connection." In turn, Gopinath and Sai (2021) provide a comprehensive definition of the smartwatch as "a wearable wrist smartphone embedded with computational capabilities and sensors for monitoring and tracking of health and fitness parameters; provides tracking information to the users and other smart devices through wireless communication networks; performs location tracking and also has a clock function." In other words, the smartwatch is a smartphone that is worn on the wrist. The so-called fitness tracker is a kind of wearable device that is closely connected to the smartwatch and, as the name implies, specializes in monitoring various health and fitness metrics. Fitness trackers, in contrast to smartwatches, often contain technology that is less costly, fewer sensors, and a restricted interface for presenting information than smartwatches. So, one might think of fitness trackers in the same vein as

smartwatches but with less functions. Finding a clear line of demarcation between fitness trackers and smartwatches is challenging, particularly as a result of the fast development of accompanying technologies and the product improvements that go along with them. In light of this fact, the term "smartwatch" is being used to refer to fitness trackers as well in this research [32].

2.5 OBSTACLES PRESENTED BY IOT

The Internet of Things is up against a variety of obstacles, many of which are still preventing further development of connected devices and applications. On the one hand, there are still a variety of issues and impediments, which particularly impede consumer acceptability. On the other hand, there are still a number of technical hurdles to overcome, which severely limits the use of IoT. These technical hurdles are also responsible, at least in part, for the limited customer adoption of the product. Because of this, a distinction is established between aspects that are essentially psychological (such as customer approval) and ones that are purely technical (technological issues).

2.5.1 CONSUMER ACCEPTANCE

One of the most difficult problems is getting people to adopt Internet of Things devices, as was indicated in the beginning. Concerns about one's level of privacy and the amount of work required to install Internet of Things devices are especially relevant from a psychological point of view for the widespread adoption of such devices [33]. Concerns about users' privacy are likely to be the most significant barrier to the broad adoption of Internet of Things devices. Consumers are prompted to do so as a result of the copious amounts of data acquired by sensors and cameras that are associated with Internet of Things applications. The fact that the technology is still in its infancy gives rise to these worries, as well as the fact that the accessibility of data by businesses as well as governments is sometimes not understandable, and the data storage is seen as being insecure. This issue may be shown with the use of technologies such as the voice assistant Alexa and RFID tags. Amazon has said in a court case that recordings and transcripts of conversations with Alexa are kept on its servers and that they are accessible to Amazon (Sauer, 2017). There is currently no way to determine the answer to the question of when the voice assistant is engaged and when the surroundings is captured. RFID tags, also known as radio frequency

identification tags, are small tags embedded in common items like cellphones that enable individuals to be tracked and may reveal more information about a person's activity than they are comfortable disclosing. The work required to set up is another significant obstacle to overcome. The installation of Internet of Things systems and devices often takes a significant amount of time and calls for a high degree of technical competence. In certain instances, the setting up of the system even requires the assistance of professional service providers, which may result in significant extra expenses [34].

2.5.2 TECHNOLOGICAL ISSUES

About the technical side of things, there are additional difficulties associated with technological problems. Data security, a lack of interoperability, high expenses, and limited access to Wi-Fi are among them. From a psychological point of view, data security is not the only issue that Internet of Things devices face. Data no longer stay confined to a single device because to the proliferation of linked gadgets and networks; instead, it moves across both public and private computer systems. Since there are so many different ways that Internet of Things devices and operating systems may be designed, there are also many different ways that privacy might be breached [34].In addition, there is an issue caused by the many distinct methods of data collecting, which prevents interoperability [35]. There is presently a lack of consistency and standards that would allow better connection and deployment due to the varied nature of the devices that make up the Internet of Things [36]. As a consequence of this, the data coming from various sensors can be measured differently, which might then lead to inaccurate or inconsistent interpretations [37]. One of the most significant obstacles to the widespread implementation of IoT systems is the expense involved. As a consequence of this, Internet of Things devices remain out of reach financially for many homes and social strata, which continues to be a significant barrier to their widespread adoption. On the other hand, technological advancement and falling manufacturing costs can make purchase expenses more affordable in the near future. Last but not least, the availability of Wi-Fi in residential areas is an issue when seen from a global viewpoint. For instance, the percentage of people in developing nations who have access to the internet at home ranges from 65 percent in urban areas to just 28 percent in rural regions. On the other hand, this issue is not often a concern in nations that have undergone industrialization. Additional technical challenges cited in the research as being obstacles to

the widespread use of Internet of Things devices include inadequate energy storage, cumbersome device sizes, a restricted number of sensors, and the need for improved algorithm design.

2.6 IOT USAGE

The utilization of IoT at the present time is an essential foundation for evaluating the acceptability of IoT gadgets. As a result of this, the next chapter will provide a high-level summary of the IoT device distribution around the world. After that, a presentation on the dissemination of IoT devices across the major application sectors in Germany will take place.

By the year 2030, it is anticipated that there will be an increase to a total of 25.44 billion linked devices throughout the globe. This is a more than doubling of the current number when compared to this. Estimates put the percentage of gadgets belonging to the consumer market at sixty percent. In recent years, there has also been a substantial rise in the number of wearable electronic gadgets that are linked to the internet. Despite the fact that the number was still at 325 million devices in 2016, analysts anticipate that the one billion device threshold will be surpassed in 2022.

A poll conducted in 2020 revealed that 37 percent of people in Germany use at least one smart home application, and this percentage is expected to continue to grow in the coming years. Especially common examples are radio-controlled outlets, radio-controlled roller shutters, consumption meters, robotic vacuum cleaners, and smart lighting. Robotic vacuum cleaners and garden equipment are also becoming more common. In a comparison throughout Europe, the percentage of internet users who make use of voice assistants is lower than the norm. This is the case when looking at use rates.

In 2022, the penetration rate of smart home appliances in Germany was 12.4 percent, and it is anticipated to reach 38.6 percent by 2026.In terms of the entertainment industry, smart TVs had a 93 percent market share of all televisions sold in 2021, and this percentage was expected to continue to rise. Moreover, game consoles enjoy a significant amount of popularity in Germany. A total of 2.3 million gaming consoles are purchased annually on average. This is a stunning statistic when one considers the country's population of 83 million people as well as the fact that such gadgets have been in use for some years. The game consoles are also commonly utilized (73 percent of users), outside of the realm of

conventional gaming, for intelligent applications. In addition to the many forms of entertainment technology that were just discussed, there were also smart speakers present in 11% of all German homes [36].

In 2019, 24 percent of Germans possessed a smartwatch or fitness tracker, while another 26 percent expressed interest in using one of these devices. In terms of the use of diagnostic applications for the Internet of Things, there was a rise from 6 to 11 percent between the years 2016 and 2020. This Internet of Things device is an excellent research opportunity for the study of IoT acceptability factors since smartwatches and fitness trackers are currently used by a very large number of people and there is a lot of interest in them. In terms of the distribution of the population, one of the most noticeable characteristics of German families is that 47.1% of people who use smart homes have a high income (Statista, 2022h). In addition, 31.7 percent of users are between the ages of 25 and 34, making up a significant chunk of the user population. This leads us to believe that factors like as age and wealth may also play a role in the distribution of other Internet of Thing's devices.

2.7 THREATS AGAINST IOT

When people speak about the Internet of Things, one of the major issues that comes up most of the time is about how secure the various IoT devices are. The absence of standards is the source of the most prevalent worry, which translates to the fact that it is up to the device's creator to determine how safe the finished product will be. Since Internet of Things devices often have uncomplicated designs and are primarily focused on the job of gathering and transmitting data, security is sometimes neglected in these devices. The development and manufacture of the product may take longer and cost more if extra security elements are included in the design. This results in their often being overlooked. On the other hand, if they are not there, the gathered data that contains personal information presents a desirable target for potential adversaries. Since they are tied together and often have no protection at all, these devices are susceptible to a significant number of assaults. This chapter will begin by describing the structure of IoT devices. After that, it will proceed to discuss typical threats while concentrating on each layer individually.

2.8 THE ORGANIZATION OF THE IOT

There are three distinct layers that make up the architecture of Internet of Things devices, data storage centers, and the network that connects them [2].

a. The Device Layer, often referred to as the Perception Layer, is comprised of devices that collect data from their respective environments (measuring physical parameters, sensing other devices, etc.). Physical assaults on the device itself may be directed at this layer with the goals of causing damage, gaining access to the information that is stored, or modifying the information in an unexpected manner.
b. Network Layer: is used for connecting to other smart devices, as well as for preprocessing, aggregation, and transfer of measured data by the Edge Gateway into the cloud. Moreover, this layer is responsible for enabling communication between the Edge Gateway and the cloud. Due to the fact that the protocol that is used to connect to the cloud functions at this layer, it is feasible to launch network-based assaults (gaining remote control, denial of service, attacks on encryption).
c. The Application Layer is responsible for completing the data processing, storing the data in a database, and providing services to the end user. The software that is operating in this layer often comprises vulnerabilities related to poor practices and social engineering rather to vulnerabilities directly related to attacks themselves. Despite this, there are still a few software vulnerabilities that need to be investigated (dictionary attacks, SQL injections, cross-site scripting). In the sections that follow, we will discuss the many kinds of attacks and vulnerabilities that might be exploited to get data that is processed and stored inside Internet of Things devices. While there are other types of attacks, such as those that disable devices (for example, battery draining, sinkhole attacks, or Denial of Service), the majority of the following attacks focus either on extracting secrets from the device or on an illegitimate increase in privileges. While there are other types of attacks, such as those that disable devices, the majority of these attacks focus either on extracting secrets from the device or on an illegitimate increase in privileges.

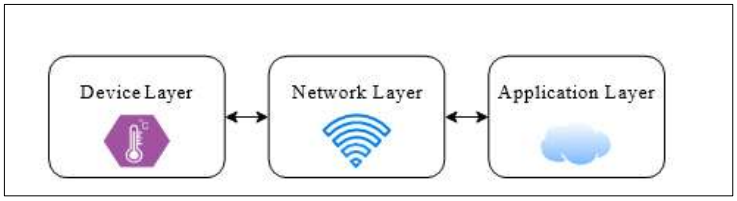

Figure 2.6: Architecture of IoT Devices.

2.9 PHYSICAL ATTACKS

The device layer is the target of physical assaults, which attempt to extract or change data there. They may be brought about either actively or passively and need access to the gadget itself in order to take their place. Active attacks, which involve tampering with the device, involve purposefully inducing a mistake to the system, either by modifying the system's environment (voltage, temperature), or by making malicious changes to the data that is stored on the system, and then observing how the system responds to the error. The observed system may then be coerced into revealing secret data or the fundamental concepts behind its inner workings, or it may even be used to launch attacks on other devices connected to the network. Attacks of this kind depend on information that has been compromised while the system is functioning normally (timing, power analysis). Since they do not alter the functioning of the gadget in any way, they are more difficult to identify [3]. The sections that follow provide many examples of different kinds of assaults that are directed at accomplishing these objectives.

2.10 EXTERNAL INTERFACES

The use of the device's external ports as a point of entry for a physical assault is one of the options available. During the manufacturing process, some debug ports, such as JTAG, UART, or other general or particular purpose inputs and outputs, are occasionally maintained rather than being permanently deactivated. This is the case in certain cases. It might be for the aim of identifying the problem and fixing it [4]. Since communication protocols exist, it is possible to use these interfaces to get access to the system itself, including the command shell, the system logs, the reading and writing of register and pin values, and the debugging of the software or firmware that is now being executed. Nonetheless, an attacker has the same possibility to get access to the system when open ports are present. He may make an attempt to debug the system and acquire unauthorized access to the information that was previously stated. The cold boot assault is one kind of attack that may be carried out on open ports. After a period of time during normal operation, the adversary causes the device to be forced to restart and boot from an attached USB flash drive or SD card that has malicious firmware [4]. This malicious version of the firmware then dumps the contents of RAM to USB memory immediately after the boot procedure. This is done because it has unfettered access to the device in question. Taking into account that the

RAM is utilized to store sensitive data while the calculation is being performed, the data that is included inside it may be examined in order to discover keys and other private information [5]. These kinds of assaults take extremely little work since there are development kits readily accessible that can be linked to debug ports. On the other hand, they are not particularly prevalent because it is simple to avoid them [4].

2.11 SIDE-CHANNEL

Attacks that use side channels go about breaching the security of a device in a roundabout manner. They measure it from the outside in an effort to get information that may lead to the discovery of a correlation between the device's inner workings and the data that is processed. In general, these kinds of assaults are broken up into two phases, which are as follows [2]:

a. Information Acquisition refers to the monitoring of the device and the recording of side-channel data, both of which imply a correlation between the actions taken by the device and its physical characteristics (such as its power consumption, electromagnetic emission, and the amount of time needed to perform a specific task).
b. Operations specific to this information is recorded by the attacker, who then does a correlation analysis with it.
c. Correlation Analysis is an attempt to compute the strength of the relationship between measured side-channel information and properties such as key length, performed operation, or the number of cache hits. This analysis is performed in order to determine whether or not there is a correlation between the two sets of information.

There are several varieties of side-channel assaults, most of which may be classified according to the kind of characteristic that is recorded. Attacks based on timing take advantage of the amount of time needed to complete a task. Due to the fact that the processing of various inputs requires varying amounts of time, this time might expose sensitive information about the cryptosystems. In addition, conditional statements, cache hits and misses, and processor instructions that do not have a predetermined amount of time to execute are shown. Then, it will be feasible to uncover essential parameters of the cryptographic algorithms, such as exponents, factors of keys, and even the value of certain bits of keys [6]. Cache timing attack is a good illustration of a timing attack that is often utilized. This attack takes use of the fact that the position of the data in memory is dependent on the amount of time it takes to access the memory to get the data [4]. Cache hits take less

time to retrieve than cache misses do. It is conceivable to link the accessible value with the number of hits and misses if the value being obtained is dependent on confidential information. The active form of the cache timing attack involves overwriting the contents of the cache and compelling the computer to reload its previous version, in contrast to the passive form, which just involves observation [7]. Simple Power Analysis (SPA) attempts to determine the algorithm that is being carried out by correlating the amount of power used by the device to the kind of command that its central processing unit (CPU) is carrying out [4]. To be more precise, the operations that are targeted by these assaults include multiplication, per-mutation, and the number of rounds. After this, SPA is able to get either the algorithm's key or the parameters that it uses [3].

The Differential Power Analysis (DPA) is an enhanced version of the Simple Parallel Analysis (SPA). The DPA makes use of statistical analysis; the same processes may have varied power usage dependent on the numerous inputs. The statistical difference may be used by the attacker, after numerous attempts, to estimate particular bits of the key without needing to have prior knowledge of the inner workings of the system [3].

2.12 INJECTION OF FAULTS

Fault injection is a kind of physical assault that modifies the device's surrounding environment in order to coerce it into behaving in an unexpected manner. Fault injection attacks deliberately interfere with the device, in contrast to side-channel assaults, which opt for a more passive variant of the attack. The injection of faults may be accomplished by the use of voltage glitching, fast temperature change, or even electromagnetic interference. This might lead to mistakes that destroy data or cause the device to miss particular commands, both of which could result in a reduction in the level of security provided by the device [8]. An attack known as voltage fault injection modifies the voltage on the power supply in order to cause errors in the way that instructions are carried out. Moreover, it has the capacity to generate a side-channel that is capable of leaking confidential data [8]. Even without taking the item out of its packaging, it is possible to launch an electromagnetic fault injection attack against it. It is used for the purpose of causing timing defects in integrated circuits, as well as for altering the input of flip-flops, bit-sets and bit-resets, and manipulating ground and voltage [9]. It has been shown that fault injection attacks are able to circumvent security features, and it is even feasible to extract the firmware on existing protected devices in an

effective manner [10]. It is important to take these assaults seriously since there is currently specialized equipment available and they don't cost a lot of money.

2.13 INCURSIONS INTO NETWORKS

A few network assaults concentrate on the process of extracting private data, as opposed to those that concentrate on disrupting the availability of devices on a network (such as a denial of service or sinkhole attack). Yet, there are some that ought to be brought up. Attacks on networks may depend on the connection protocol that is being used (RFID, NFC, ZigBee, Bluetooth, or WiFi), but the ideas behind these attacks all follow similar patterns.

2.14 ACCESS DENIED WITHOUT PERMISSION

Accessing systems without any kind of authentication (like RFID) [11], using default credentials, or attacking systems that communicate keys in an unencrypted manner (like ZigBee) [2], are the three simplest ways to launch an assault.

2.15 EAVESDROPPING

When an attacker is conducting eavesdropping, they are attempting to gather relevant information by capturing data as it is being sent [12]. The majority of RFID systems and NFC devices do not have enough memory to provide encryption in all circumstances [11], which means an attacker may get access to sensitive information with little effort on their part. During an attack known as "Man in the Middle," the perpetrator acts in a manner similar to that of eavesdropping by observing the conversation in an effort to get unencrypted data. [6] Moreover, he has the capability of altering the contents of communications before they are sent to their final destination [11].

2.16 SYBIL ATTACK

The Sybil attack is a kind of distributed denial of service attack that involves the creation of many bogus identities for a single node [13]. Using them, it is able to circumvent voting systems [6], redundancy mechanisms [11], attack routing algorithms [11], and, more broadly, impair the security of the network as well as the integrity of the data. An additional capability of the Sybil assault is to spread malware to websites in order to steal information [13].

2.17 ATTACKS MADE USING SOFTWARE

Application layer assaults are the most common kind of software attack. Attacks often concentrate on getting access rights or authenticating themselves in this layer since the software that runs in it is so varied. As there is a considerable lack of standardization of security (mostly owing to the variety of IoT systems), vulnerabilities are often related to poor practices [14]. This is the case because there is a substantial lack of standardization of security.

2.18 MALWARE

Malware is a malicious program that is used to infect applications with the intention of changing the behavior of the system, gaining unauthorized access to information, or simply disrupting its normal process [11]. Malware is a program that is designed to cause harm and is used to infect applications. Malicious software comes in a wide variety of forms, including computer viruses, worms, trojan horses, spyware, ransomware, and adware. Software updates and email attachments are the most common entry points for malicious malware into a computer system. Then it has the ability to duplicate itself and propagate over the network. A firewall, anti-virus software, and detectors are often included with standard protection [6].

2.19 MALICIOUS SCRIPTS

In addition, nefarious programs provide another entry point into the system. IoT systems may be susceptible to code injection attacks (due to API flaws and poor security procedures [11]), which allow the attacker to execute code in the browser of the victim [12]. This vulnerability is caused by API vulnerabilities. When being put into action, this malicious code has the potential to provide the attacker with access to the IoT system as well as an increase in his privileges [6].

The term "SQL Injection" refers to a kind of attack known as "code injection," in which a piece of executable code is inserted into a SQL query and then run across a SQL database. The attacker may use this method to run queries and instructions over the database if the query is not sanitized. Input validation provides a simple defense against this kind of assault. SQL injection is rather common, however, since this process is often not carried out in the right manner [15].

Cross-Site Scripting, often known as XSS, is another method that may be used to inject code into the system for the purpose of execution. However, unlike SQL Injection, XSS modifies the source code of the IoT application itself. An attacker may use XSS to their advantage to steal credentials and acquire root access to the system. Data validation is another method that may be used to avoid this [15].

3. SOLUTIONS FOR IOT ATTACKS

Due of the assaults described above, there has to be an inbuilt method of protecting the devices that are utilized for the internet of things. As a result, a pattern of solutions based on both hardware and software has evolved as a viable option. They vary depending on the architecture of the device, the purpose for which it is designed to be used, the vendor that is supplying the solution itself, or the kind of asset that has to be safeguarded. Nonetheless, several of the proposed remedies turned out to be more useful and widespread. Although there is a wide variety of potential software-based solutions, ranging from specialized cryptography libraries to development kits, the use of hardware seems to enhance these solutions in a variety of ways.

To begin, software has to depend on the speed of the central processing unit. The Internet of Things devices aren't always extremely dependable, especially in many circumstances. It's possible that this will render the software solutions now in use completely ineffective. Hardware solutions, on the other hand, are often better suited to their assigned tasks and may easily exceed any software.

The second issue is how well one can defend themselves against assaults. A significant number of physical assaults don't even involve the software running on the device that's being attacked (such as previously mentioned side-channel attacks). Hardware solutions often come pre-installed with their very own integrated cryptographic keys, as well as tamper-resistance and, in some instances, the ability to limit access to their memory. They have a higher resistance to damage.

The third and last point that has to be brought up is the fact that the hardware gives us the ability to protect the device even before the device's software is loaded for the very first time. The idea of trust comes into play at this point in the conversation. Trust instils in us the conviction that technology will carry out its functions in the manner that we anticipate and that it has not been corrupted [16]. There is something called a static trust, and its purpose is to ensure that the system of the device is secure when it is initially turned on (from firmware to loading of the OS). The second kind of trust is called dynamic trust, and it describes the degree to which a system may be trusted during the whole of its execution. These ideas will be discussed in more depth in a later section.

In this chapter, we will examine the two hardware-based options that are used most often for Internet of Things devices. The first component is a Trusted Execution Environment, which

provides protection for the programs on the system by isolating them and the data they store from access by other parties. The second component is a Trusted Platform Module, which focuses primarily on the generation of keys and the maintenance of their safety.

3.1 EXECUTION ENVIRONMENT THAT CAN BE TRUSTED

The Trusted Execution Environment, often known as TEE, is a tamper-resistant processing environment that allows a program to be run in a safe setting while remaining separate from the rest of the computer. TEE may be approximated by using a separation kernel. As compared to a traditional kernel, a separation kernel's purpose is to split the operating system into distinct subsystems, each of which often possesses a unique degree of protection. Not only does the kernel break the system up into partitions, but it also ensures that these partitions remain distinct from one another (in both time and space) and regulates the flow of data between them (if this flow is even allowed) [16].

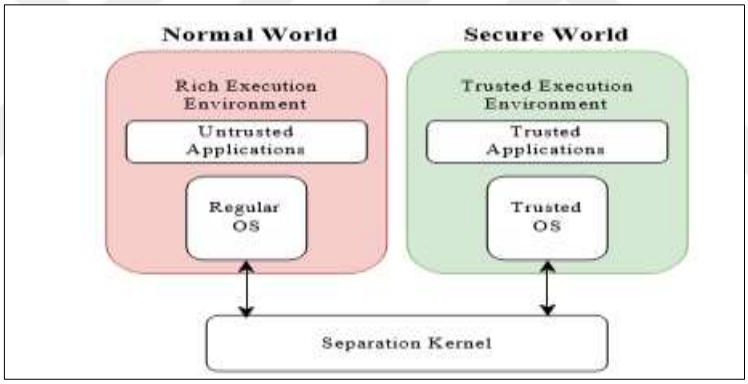

Figure 3.1: System with Trusted Execution Environment.

TEE ensures the validity and secrecy of the code and data it contains, in addition to providing resistance against software and hardware assaults [16]. This is made possible because to its capabilities, which include separated execution and secure storage. The construction of two distinct execution domains, referred to as the Normal World and the Secure World, forms the foundation of TEE [17]. The so-called Rich Execution Environment is where programs are run, and Normal World acts as the environment for this (REE). Both trustworthy and untrusted programs are included in REE. The operating system (OS) is a standard one. In spite of the fact that this enables REE to run a wider variety of programs, it also leaves it more open to the possibility of being attacked. The Trusted Execution Environment (TEE) is supported in the Secure World environment. Trusted Apps will operate from inside the

Secure World domain (TAs, also called Trustlets). Since trustlets are in charge of activities that are particularly sensitive to security, [18] they are kept safe by the use of software and cryptographic isolation.

Each of the two worlds, Secure World and Normal World, has its own collection of peripheral devices, drivers, registers, and cache [17]. Even though Secure World is permitted to utilize the resources of Normal World, direct access from Normal World is not permitted under any circumstances. TEE is a notion that has been adopted by a number of different companies and included into the solutions that they provide. This thesis will discuss Arm Trust- Zone and Intel's Software Guard Extensions, both of which are examples of hardware technologies that provide support for TEE implementations.

3.1.1 Trust Zone

Trust Zone is one of the various implementations of TEE. It provides ARM architectures with hardware security features [19]. Trust Zone now supports Cortex-A processors (architectures Armv7-A and Armv8-A) as well as microcontrollers based on the Cortex-M family (architecture Armv8-M) [20]. The following describes the distinctions that exist between Trust Zone for Cortex-A and Cortex-M: The Cortex-A architecture has a protected monitor mode by default. The Secure Monitor is used in the process of context switching between the Normal World and the Secure World. Normal World requires certain criteria to be met before switching, such as a hardware interrupt, an abort signal, or a particular instruction [19]. Secure World allows for unfettered switching. Since the secure monitor is not used in the Cortex-M architecture, switching-induced delay is reduced. Instead, the degree of security is defined by the code that is being run, and more precisely, where in memory the code is located [19][20].

a. Non-Secure Bit: While addressing, Trust Zone makes use of an additional bit. The Non-secure (NS) bit identifies which of the two possible worlds—Secure or Normal—is now attempting to access memory [21]. This tidbit gives a means of checking for access that has not been permitted. In addition, the value of the NS bit for the current domain may be located in the Secure Configuration Register [20]. Both Secure World and Normal World use distinct page tables to conduct their operations. As compared to Secure World, Normal World is only able to produce requests for memory that has been labeled

as non-secure. On the other hand, Secure World is able to generate requests for both Secure and Non-secure memory [4].

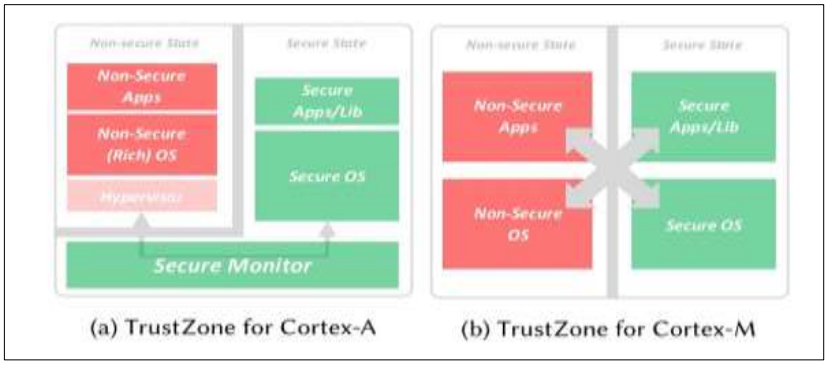

Figure 3.2: Trust Zone Difference Between Architectures [20].

When addressing, any module that is aware of Trust Zone will utilize the NS bit, whereas modules that are not aware will have their bit set to 0 (representing the Normal World) by the processor. Caches have the capability to store utilizing addresses that still have the NS bit attached, which splits them in effect. By the use of controllers such as the Trust Zone Address Space Controller (TZASC) or the Trust Zone Memory Adapter (TZMA), it is possible to partition memories that are unaware of their surroundings [4].

b. Protected Gateway: As was discussed in section 2.1, Secure World is free to use the resources that are available in Normal World. Nonetheless, in the event that the Normal World wishes to make a secure code call to the Secure World, then doing so must be carried out appropriately and stored in a pre-defined area. The process of calling between secure and non-secure software is managed with the help of instructions that are kept in a dedicated non-secure callable memory (NSC) [22]. The following are the instructions:
c. Secure Gateway (SG): must transition to the secure state upon receiving the first command from the secure entry point.
d. Branch with exchange to non-secure state (BXNS), often known as branching or returning to non-secure software from Secure software.
e. Branch with link and exchange to non-secure state (BLXNS) – use secure software to call non-secure functions.

3.1.2 Sau and Idau

The security attribution unit, often known as SAU, or the Implementation Defined Attribution Unit may be used to determine whether memory is secure (IDAU). Both have a set of registers that may be customized, and using those registers, they can assign Secure, Non-secure, or non-secure callable to addresses that come from memory. One of the components of the CPU is referred to as the SAU unit. Whenever the device is reset, the SAU is rendered inoperable and requires that it be configured by software. It is essential that this program be used inside a secure environment. Memory areas that may be established by SAU are not predetermined in any way, and both their number and range are up for grabs [20]. The IDAU is a hardware device that is fixed and located outside of the CPU. While it has many of the same capabilities as SAU, its settings as well as its memory map are hard-coded and cannot be modified by the program. IDAU is also going to remain active after the reset by default, but you have the option to turn it off [20]. One of the conceivable applications is to use IDAU to specify the security properties of the memory map and then use SAU to overwrite certain sections of the memory. This is one of the possible applications. When SAU and IDAU are both used to describe the same memory sector, whichever security level provides a better degree of protection will be used. There is also the chance that SAU will have memory regions specified as zero or that the software will never activate it, and there is also the possibility that IDAU will not be provided with the CPU or will be disabled. Since neither the SAU nor the IDAU has provided a definition, the whole memory space has been declared to be secure [20]. Memory Protection Unit (MPU) is what makes it feasible for system software (like the OS kernel) to monitor transactions directly on the bus and identify any breaches [23]. SAU and IDAU are responsible for assigning different degrees of protection to different areas of memory. Moreover, MPU allows system software to specify its own access rights for the sake of security [24]. When there are several security accesses configured for the same address, the one with the greatest priority will be picked. AMBA Bus Trust Zone is an implementation of the Advanced Microcontroller Bus Architecture (AMBA), which is used to impose security limitations on slave components (memory and peripherals) [19]. The Advanced Peripheral Bus (APB) is an interface that has a low power consumption and a limited bandwidth. Via the use of a bridge, APB is able to link the masters of external peripherals, known as slaves, to the system bus. It does not do a check for security permissions; rather, that task is carried out by the

bridge itself in order to maintain backwards compatibility (since APB is a relatively straightforward protocol) [19]. The Advanced High-performance Bus (AHB) or the Advanced extensible Interface (AXI) protocol is what forms the backbone of the communication between the CPU and the peripheral devices that are connected to the main system bus. As was discussed earlier in this chapter, the NS bit is used by the Trust Zone operating system to indicate the kind of memory to which read and write operations are being directed [19]. Extended address lines are used by the system bus in order to make room for this functionality [4]. A matrix configuration for the bus is possible given the possibility of many masters and numerous slaves. The previously stated APB bridge, Memory Protection Checkers (MPC), and Peripheral Protection Checkers are the very last components that may be found between the bus and the memory and peripherals (PPC). These are Trust Zone-aware peripherals that can act as gates and verify access rights in addition to providing security characteristics to the network. PPC is responsible for assigning and checking the security and privilege levels for peripherals, interrupts, and pins. These levels may either belong to the User or have Privileged access. The same applies to memory blocks when using MPC. Moreover, NVIC Trust Zone includes an expansion for the Nested Vectored Interrupt Controller (NVIC). The typical use of NVIC correlates with giving varying priority to interrupts, which enables the handling of crucial ones first and also allows for the chaining together of interruptions [25]. Interrupts may have their respective security levels adjusted to either Secure or Non-secure when using Trust Zone [20].

3.1.3 Extensions for the Software Guard

Software Guard Extensions (SGX), which is a further example of an implementation of Trusted Execution Environment, additionally protects the confidentiality and integrity of both the code and the data. It accomplishes this goal by storing them in enclaves that are secured by hardware [26].

3.1.3.1 Enclaves

The enclave operates as a separate and compartmentalized execution environment. Enclaves often have their contents encoded in order to maintain their users' privacy. In addition, an enclave is guarded against entry from untrusted third parties [27], which means that one can only enter it by following very specific instructions. During the establishment of the enclave,

a cryptographic hash is generated. This is done so that the content's integrity may be checked before the first load of material that is not trusted [4]. This approach will be unsuccessful if the value that is seen is not the same as the one that was anticipated. A further step in the process of attestation may include the use of hash, which is also known as measurement. The hash of the material is used in the software verification process (such as code, data, or keys). A hash is produced by the attestation key of trusted hardware, which demonstrates that the program is safeguarded by the hardware and is operating in the environment it was designed for. It is possible at any moment for third parties to make a request that the content of the enclave be authenticated, in addition to having its integrity checked [4].

3.1.3.2 Organization of the memory

As can be seen in Figure 2.3, the memory of SGX has numerous hierarchical levels of structure. Main DRAM sets aside some of the memory to be used as Processor Reserved Memory (PRM). This section of memory stores anything that is shielded from access by untrusted sources. Enclave Page Cache (EPC) is a subset of PRM, whereas the remaining portion of PRM is used to store software that is not related to enclaves (for example, system software). The EPC is a shared resource that has pages for each enclave that is currently operational and is used by all of them. In addition, each and every bit of data is encrypted using keys that are kept in the processor [27].

The Enclave Page Cache Map (EPCM) is where the condition of each page is kept [4]. It contains the page's validity as well as its kind (normal or special types, such as SECS pages), as well as the identity of the enclave that is the owner of the page itself [27].

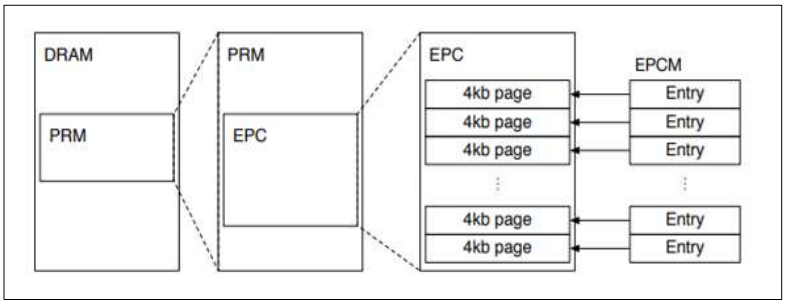

Figure 3.3: Sgx Dram Organization [4].

While EPC is composed of pages, the following control structures may also be found inside its content:

a. The SGX Enclave Control Structure, or SECS, stores the information for a single enclave. This metadata includes the hash and the size of the enclave [4]. Since SECS is generated and destroyed at the same time as the enclave it represents, it is inextricably linked to the existence of the enclave.
b. One thread is embodied by the Thread Control Structure (TCS) for each enclave that is defined. There may be several TCS constructions owned by an Enclave [27].
c. The Save State Area (SSA) is used for the purpose of saving context while interrupt procedures are being handled [27].

3.2 SECURED HOST PLATFORM MODULE

The Trusted Environment Enclave (TEE) is a processing environment that is located within the kernel. The Trusted Platform Module (TPM), on the other hand, is built on specialized hardware [28] that is coupled through a straightforward interface. Its primary functions include safe booting, the performance of cryptographic operations, the management of keys (creation and storage), attestation, and measurement of the software device to which it is attached [29].

3.2.1 The Foundation of Trust

The Trusted Platform Module (TPM) is based on the principle that software cannot be trusted automatically. Because of this, the Trusted Platform Module (TPM) includes a hardware-based security component known as Root of Trust (RoT). Due to the fact that it serves as the basis. Due to the fact that this would further erode users' faith in the platform, it is imperative that it be safe by default. There are many different certifications that may be used to determine the degree of safety provided by RoT [30], and each of them is broken out into further detail in Section?? Regardless of whatever option is selected, TPM is able to demonstrate this level using a Certificate of Authenticity and a key known as an Endorsement Key (EK). EK is a key pair for RSA 1, consisting of both a private and public key. The secret portion of EK is never sent outside of TPM and cannot be read from any location other than inside it. Since it is the sole thing that is used for authentication [29], it is essentially the identity of the device.

RoT for Storage, RoT for Reporting, and RoT for Measurement are the three distinct varieties of RoT that are included in TPM [28].

3.2.2 Rot for Storage

The RoT for Storage (RTS) protocol makes it possible to store keys and other types of confidential data. It's possible that these delicate items are kept in a few different places that are insulated. A shielded location is a memory that can't be directly accessed from anywhere other than the trusted platform module (TPM). Certain information may be provided if it is supported by the appropriate authorization, whereas some data, such as EK, are contained inside the TPM at all times. Outside of shielded locations, items may be kept in Protected locations, which allow access to the objects but encrypt the sensitive sections of them [28]. The TPM may hold non-secret information as well. For instance, Platform Configuration Registers (PCRs) are an important component of the process of attestation, and as such, they have to be readily available [28]. PCRs include measured digests that have been saved for integrity checks (in the form of SHA-1 hashes). The use of such measurements makes it feasible to identify any potential changes in the data. Any newly acquired measures are weighed against those included in PCRs. From that point on, the third party has the ability to either affirm the reliability gained via the integrity report or deny software that does not comply [31]. There is more to creating PCRs than just writing them in. Instead, lengthening them is done. The value that is to be added and the PCR are both inputs that are brought into the Function Extend, which then creates a hash of the combined value [29]. The use of hashing each time we wish to extend gives protection against attackers who could attempt to change the state of the bits in these registers.

3.2.3 Rot for Reporting

Reports done on the content of PCRs are done by *RoT for Reporting* (RTR). RTR serves as an interface for accessing information such as the platform’s identity (EK) or digests stored inside the RTS. It also protects shielded locations from possible attacks, either software or physical [28].

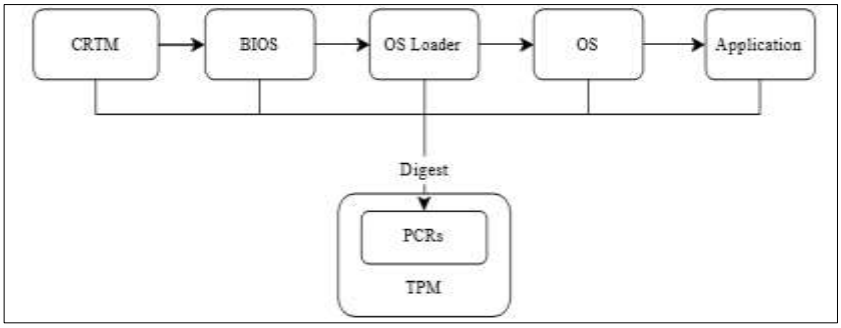

Figure 3.4: Platform Measurement Process.

3.2.4 Rot for Measurement

Measurements of the system's integrity are taken by the RoT for Measurement (RTM) component and then saved in the RTS database. While evaluations may be carried out on either the data or the program, it is critical to do an integrity check on the platform at the beginning of the start-up procedure. This not only offers security against rootkits2, but it also makes it feasible to detect any potential firmware modifications. Core RTM (CRTM) is the tool that is used to do measurements of the platform [28]. When the gadget is powered up, CRTM is the very first item to be placed into it. It then makes a report on the state of the BIOS's integrity, and if everything is in order, it executes the BIOS and gives it control of the system. The TPM, and more especially the PCRs inside it, is where each and every report is saved. The integrity of the OS loader is validated by the BIOS before it is executed, and this process continues all the way through the operating system and into the apps. Before any portion of the system can be executed, it must first get feedback from every other section [29]. Figure 4.4 depicts the whole procedure from start to finish.This chain of cooperation between TPM and CRTM creates the first Trusting Building Block (TBB), which serves as the foundation for the whole of the trusted platform and the instantiating of the Root of Trust [31].

Now that it has been initialized and is safe, the TPM is able to deliver the following features:

a. Encryption using symmetric keys • Encryption using asymmetric keys.
b. Hash engine - For creating digests for PCRs, hashing methods are necessary in order to be used. TPM contains among other implementations of SHA-13 for hashing and HMAC4 for signing and verification purposes. SHA-13 is used for hashing, while HMAC4 is used for signing.

c. RNG module - TPM does not employ fake RNG. Instead, it comes with its own entropy-generating mechanism. After that, the mixing function will get random integers from the state register and utilize them wherever they are required.
d. Key creation - For the purposes of algorithms such as RSA, the TPM is able to either create a new key from a seed value or value acquired via RNG. Both of these options are available to the user. Inside the protected confines of the TPM is where the secret portion of the key is stored safe and sound.
e. Memory - The TPM has both random access memory and volatile memory. RAM is where PCRs, the IO buffer, and data and keys that were loaded from the outside are stored. Non-volatile memory, often known as NV memory, is used for long-term storing and is made up entirely of protected places [28].

3.2.5 Trusted Execution Technology

Another technology that is working toward the provision of hardware-based security with the assistance of the TPM is called Trusted Execution Technology (TXT), which is developed by Intel. TXT provides a trusted environment that separates and safeguards running applications, which contributes to an enhancement in the platform's overall safety. It is possible to have faith that this environment will be in a secure condition right from the start, as well as measure and verify the trustworthiness of software, and store sensitive data in a safe manner [32].

TXT's architecture necessitates the storage of cryptographic keys, the authentication of software, measurements, and safe booting, all of which are vital components. According to the information presented in section 2.2, each of these technologies is a tool that can be found in the TMP toolbox. Because of this, TXT also contains a TPM chip that is coupled with Low Pin Count (LPC) bus [32]. This straightforward bus, which is also guarded by the operating system and access permissions, makes access from the outside to the TPM more difficult and, as a result, more secure [33].

3.2.6 Acm Stands for Authenticated Code Module.

When determining whether or not an environment can be trusted, it is necessary to begin measuring it as soon as the platform is powered on, even before the BIOS has been loaded. The same is true for computer software. TXT must first take measurements on its

configuration and policies, in addition to the program itself, before it can begin its execution. The Authenticated Code Module is the one responsible for carrying this out (ACM). At the time that the program is being evaluated by ACM, it is moved to a safe location where it is guarded against unauthorized access and cannot be altered. [34] The program may be run after the measurements have been completed and it has been verified that it complies with the launch policy. The BIOS and SINIT are the two distinct varieties of ACM. SINIT ACM is responsible for starting up the system software, while BIOS ACM is responsible for measuring and launching the BIOS [33]. It is necessary for there to be a safe point, which was referred to earlier as the Root of Trust. This is a hardware-based protection that is trusted by default and is used to extend this confidence to other components of the system. With the help of the measurements carried out by ACM, TXT is able to construct a chain of trust beginning with RoT. Static Root of Trust for Measurement (SRTM) and Dynamic Root of Trust for Measurement (DRTM) are the two ways that may be used in the process of constructing these chains for TXT [33]. SRTM and DRTM SRTM is begun as the first thing after turning on the device. DRTM is started as the last thing. Microcode, which verifies the correct operation of BIOS ACM, is executed by the hardware. From that point on, each individual component of the boot process, beginning with the BIOS, is measured just before it is put into action. This step generates the measurement component, which is necessary for passing as it requires that the component pass integrity tests and be secure. chain of trust across the whole platform in sequential order [35]. The measurements are saved in PCRs that are permanently placed inside the TPM. These measurements will remain there until the platform is either powered off or reset. Since their information is immutable, it is referred to be static, and it demonstrates that the whole system can be trusted. This serves as a foundation around which the platform may be constructed. Measurements of this kind are referred to as platform configuration measures [33]. As software that is performed after the boot process of the platform also needs to be measured and compared against predicted values in order to be trusted, the operating system has the ability to draw upon DRTM in order to put it into a known state [34]. SINIT ACM performs software analysis and saves the results in dynamic PCRs, which ACM is able to replace without requiring the platform to be restarted. When you reach that point, attestation or verification of the launch policy will be available [33].

3.2.7 Launch Conditions That Have Been Measured

Measured Launch Environment (MLE) or host operating system refers to the first operating system or hypervisor that is started after booting. There might be several measurements for a single MLE, for example, if there are various configurations of the same operating system [33]. If MLE has successfully passed the tests performed by SINIT ACM and launch policy, it will be able to perform the duties of an operating system in a trusted environment and provide attestation and other security-related services.

3.2.8 Virtualization

Virtualization may be used inside TXT with the utilization of Virtual Machine Monitor (VMM), which is a hypervisor made possible by Intel's Virtualization Technology [35]. The VMM is the component that is accountable for the production and operation of many instances of virtual machines. To be more explicit, every secure container operates its own virtual computer, which includes a guest operating system [4][33] Operating systems of this kind are portable between platforms and may keep their secure settings intact.

3.3 FLAWS IN THE PROPOSED SOLUTIONS

While it is reasonable to anticipate that every one of the solutions described in this chapter would perform without defect and thwart every one of the assaults against which they claim that they are immune, the fact is that this is often not the case. There may be exploitable faults in the system, and they may have been introduced either by design or by implementation. In this part, we will present a summary of the flaws that have previously been uncovered for specific solutions. That is not an exhaustive list of all the vulnerabilities that have been reported; they may be found on specific webpages.

a. Trust Zone: One strategy for attacking Trust Zone or TEE systems in general is to compromise the TEE kernel. This may be accomplished by the escalation of privileges from NW space or through the exploitation of a vulnerability that is already present in the kernel itself. It is also possible for the REE kernel to be corrupted, for instance using a susceptible TA [21]. An attacker has possibilities to succeed when they have access to an attack surface. As the surface area is increased, the number of potential points of entry into the system also increases. Because of this, software that is not essential or that

is not secure should not be executed in the TEE kernel area. One relevant example would be software drivers, which are often instances of common problems. Moreover, although comprehensive APIs might be advantageous, they can also expose an excessive number of kernel system calls to TAs [21]. If measures are not taken to prevent it, software may leak information that can be exploited via debugging channels and branch predictors. Side-channel attacks are another kind of vulnerability that might affect TEEs [21].

b. SGX: It would seem that the known vulnerabilities of SGX are side-channel attacks that are dependent on performance [4]. An access-driven cache-timing attack against AES has been shown to be effective in at least one instance. This attack takes use of Intel's Performance Monitoring Units (PMU) to offer precise timing information to monitor cache lines, and as a result, it is able to break the susceptible AES implementation that is executing inside of the enclave [36]. The Row Hammer attack [37] is another vulnerability that has been discovered in SGX systems. Row Hammer's purpose is to repeatedly access DRAM (while avoiding the cache) a sufficient number of times to change the value of bits that are located in contiguous rows [38]. If it is successfully launched against the memory of the enclave, it will cause its integrity check to fail, which will cause the processor to lock up and will need a reboot of the system [37].

c. TPM: Another solution that is susceptible to timing assaults is TPM, which has this vulnerability. It has been determined that there is a link between the amount of time it takes to execute the production of an ECC-based cryptographic signature and the key that is stored in the TPM. This approach enables complete recovery of the key and is able to penetrate both Intel's and STMicroelectronics' TPMs [39]. A desirable target for an assault, given that the measurement of the bootloader also dictates the security of the TPM. It is possible for the firmware that is loading the bootloader to be reprogrammed [4, but only if the required security precautions have not been taken.

d. TXT: DRAM attacks were able to exploit early versions of TXT, which had the same flaw as TPM. One of the attacks was successful because of flaws in the implementation. This allowed it to compromise the hypervisor (which was securely loaded). It was discovered that malicious DMA might be scheduled using graphics and audio cards [40].

4. METHODOLOGY

In this chapter we will reveal the practical part regarding the IoT field security. We use the machine learning algorithms to detect and predict the DoSS attack. Figure 5.1 is the methodology that we followed.

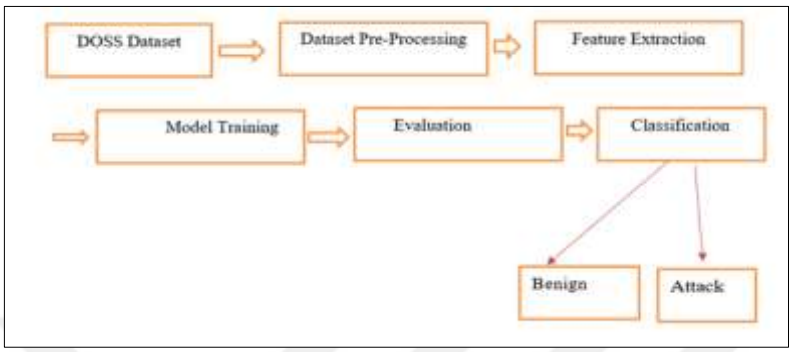

Figure 4.1: Research Methodology.

4.1 APA-DDoS Dataset

With an exponential increase of connected devices, the detection of malicious connections has become more a challenging task. Intrusion Detection Systems (IDS) is one of the typical mechanisms used for these purposes. Given the sophistication of the attacks and a variety of normal traffic patterns use of machine learning is becoming mainstream in IDS. To know the enemy face the Machine Learning (ML) tools need to have access to attack patterns that represent all sorts of attacking traffic. Unfortunately, not all patterns for infamous attacks are presented in the datasets available for the machine learning community. This project is aimed at bringing the existing gap between common attack patterns included ACK and PUSH-ACK flooding DDoS attacks that have been underrepresented in the existing datasets. The developed dataset can be used by IDS developers to increase the detection ratio by detection modules.

Column	Description
StartTime	Record start time.
TotPkts	Total number of packets in transaction.
TotBytes	Total number of bytes in transaction.
LastTime	Record last time.
Seq	Argus sequence number.
Dur	Record total duration.
Mean	Average duration at records aggregate level.
StdDev	Standard deviation of the duration at records aggregate level.
Sum	Total duration at records aggregate level.
Min	Minimum duration at records aggregate level.
Max	Maximum duration at records aggregate level.
SrcPkts	Source to destination packets count.
DstPkts	Destination to source packets count.
SrcBytes	Source to destination bytes count.
DstBytes	Destination to source bytes count.
Rate	Total packets per second in transaction.
SrcRate	Source to destination packets per second.
DstRate	Destination to source packets per second.
Category	Type of binary traffic, either Normal or Attack.
Subcategory	Type of multiclass traffic, either Normal or Attack (UDP, TCP, HTTP).

Figure 4.2: Columns Description for the Dataset.

4.2 DOS ATTACK

User identification is being done for a variety of reasons and goals, but one of the primary ones is to avoid assaults. The so-called Denial of Service (DoS) assault is one of the most well-known types of attacks that may be guarded against in this manner. An effort by a hacker to prohibit authorized users from gaining access to the information or services provided by a provider is referred to as a denial-of-service attack, or DoS attack, in the industry [9]. Figure 3.1 provides a graphical representation of this kind of assault. The goal of the attacker is to interrupt the connection by continuously disturbing the server service or network infrastructure. If the attacker is successful, the host connection will either be partly or entirely lost.

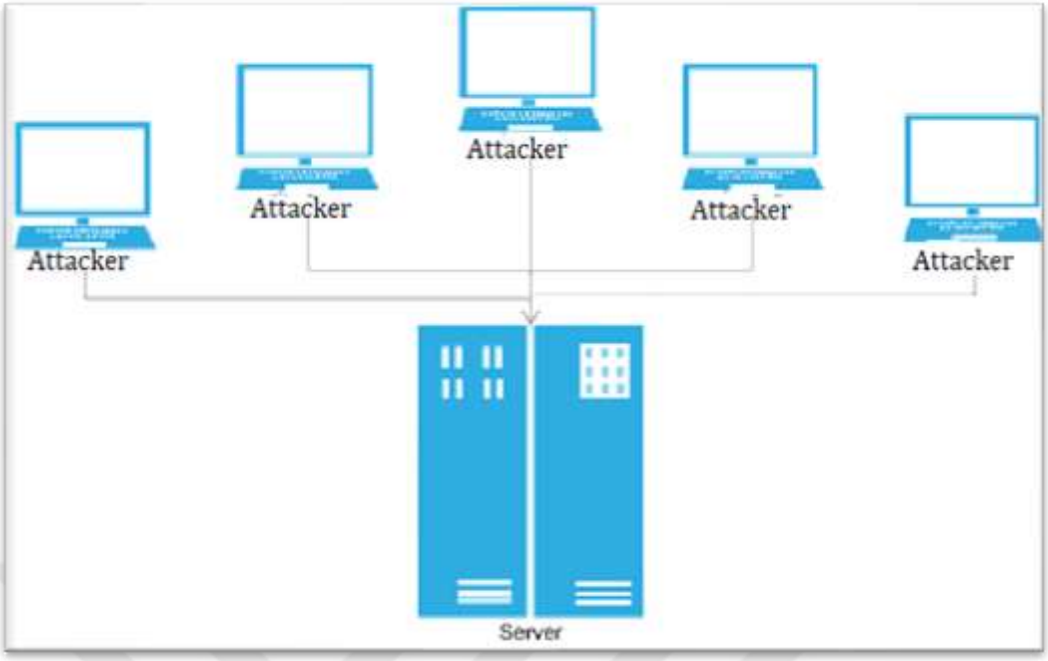

Figure 4.3: Dos Attack Scheme, Source.

4.3 PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization (PSO in short) is a swarm algorithm that is inspired by the swarming of birds or fish. this intelligence is based on a numerical optimization algorithm introduced in 1995 by Mr. James Kennedy and Russel Eberhart [46,47,48]. Social animals group together for a variety of reasons, and flocks or groups are always advantageous for the survival of individuals for several reasons:

Foraging individuals of the swarm can benefit from the discoveries and previous experiences of other individuals in the search for food.

a. Defense against a predator: a group of animals (e.g., a flock of birds) can defend itself more effectively against a predator. More eyes mean a greater chance of spotting danger, the amount of potential prey in the herd reduces the danger to the individual, and a group of animals is able to confuse the enemy. of course, this behavior also brings disadvantages, which are not simulated by PSO (e.g., weaker individuals do not die, as with the genetic algorithm). in PSO, all individuals survive and try to:
b. Proximity principle: the population should be able to perform simple spatial and temporal calculations.

c. Quality principle: the population should be able to respond to quality factors in the environment.

d. Diverse response principle: the population should not engage in the same response.

Stability principle: the population should not change its behavior style every time the environment changes.

e. Adaptability principle: the population should be able to change its behavior as long as it is advantageous in terms of computational cost.

With PSO, the solution is obtained through random search (random search), which is equipped with swarm intelligence. This means that PSO is a swarm intelligent search algorithm. The search is carried out by randomly generated potential solutions. This collection of potential solutions is called a swarm, and each potential solution is called a particle. Browsing is influenced by two ways of learning. Each particle learns from another particle, which represents the so-called social learning and at the same time from her own experiences, the so-called cognitive learning. Thus, thanks to social learning, the particle obtains the best solutions visited by any particle in the swarm, called G best (global best). Through cognitive learning, it reaches the best solution that each particle has found and visited so far, called P best (personal or particle best). Each particle has its own trajectory, called position xi and velocity vi, moving in the searched space by gradually updating its trajectory. Particle populations adjust their trajectories based on P best and G best positions. all particles have a fitness value, which is evaluated by an objective function designed for optimization. the particles move in the solution space by following the optimal particles. The algorithm initializes a group of particles at random positions and then searches for optima by updating generations. in each iteration, the particle is updated based on the two best positions P best and G best. in d-dimensional space, the position and velocity of the i-th particle in the time step t are represented by a d-dimensional vector, $xt = (xt, xt, \ldots, xt)\ T$ and $vt = (vt, vt, \ldots, vt)\ T$.

5. RESULTS

In this section we will reveals our experiment results from machine learn as the figures reveals.

```
Runing Logistic Regression algorthim....
0.7694444444444445
              precision    recall  f1-score   support

      Benign       1.00      1.00      1.00     22552
    DDoS-ACK       0.54      0.52      0.53     11411
DDoS-PSH-ACK       0.54      0.56      0.55     11397

    accuracy                           0.77     45360
   macro avg       0.69      0.69      0.69     45360
weighted avg       0.77      0.77      0.77     45360
```

Figure 5.1: Logistic Regression Algorithm.

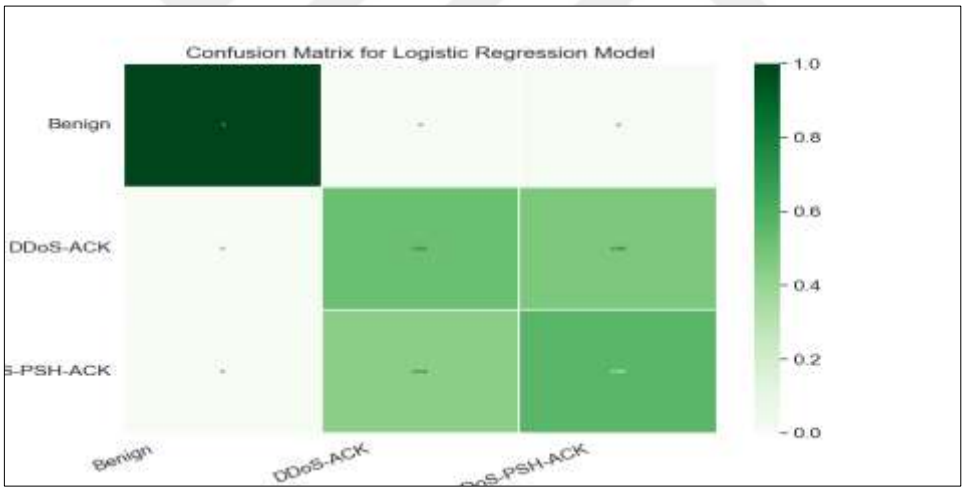

Figure 5.2: Confusion Matrix for Logistic Regression Algorithm.

```
------------------------------
Runing KNN algorthim....
0.7901455026455027
              precision    recall  f1-score   support

      Benign       1.00      1.00      1.00     22552
    DDoS-ACK       0.59      0.57      0.58     11411
DDoS-PSH-ACK       0.58      0.60      0.59     11397

    accuracy                           0.79     45360
   macro avg       0.72      0.72      0.72     45360
weighted avg       0.79      0.79      0.79     45360
```

Figure 5.3: KNN Algorithm Results.

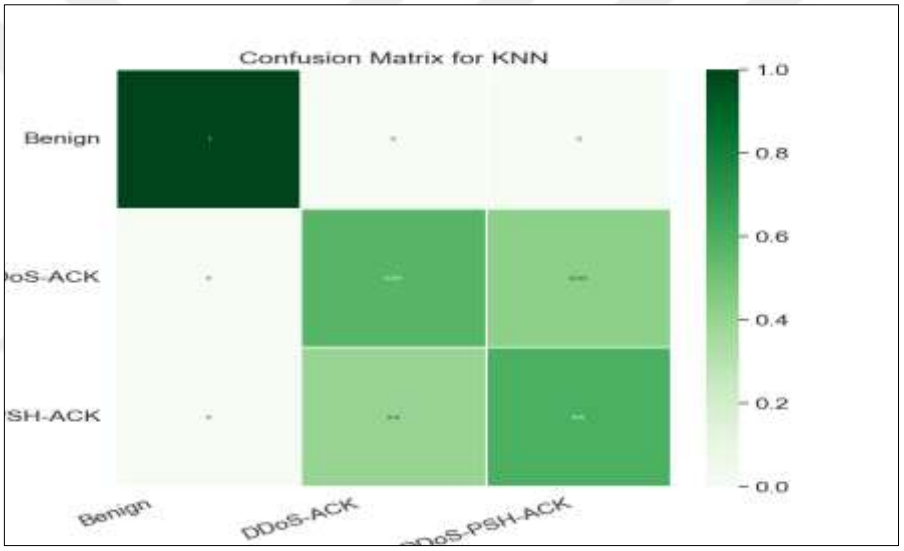

Figure 5.4: Confusion Matrix for KNN Algorithm

```
              precision    recall  f1-score   support

      Benign       1.00      1.00      1.00     22552
    DDoS-ACK       0.50      1.00      0.67     11411
DDoS-PSH-ACK       0.00      0.00      0.00     11397

    accuracy                           0.75     45360
   macro avg       0.50      0.67      0.56     45360
weighted avg       0.62      0.75      0.66     45360

the end ------------------------------
```

Figure 5.5: SVM Algorithm Results

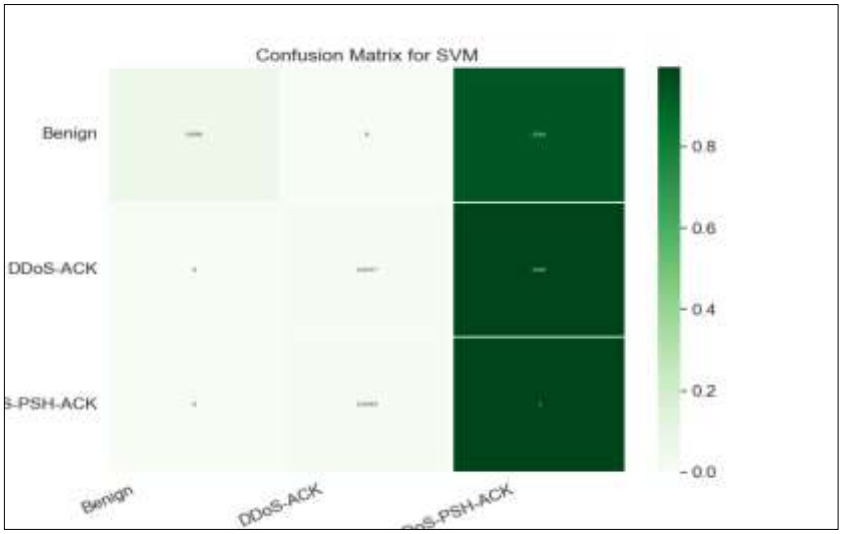

Figure 5.6: Confusion Matrix for SVM Algorithm.

6. CONCLUSION

The first section of this thesis surveyed the history and current state of the infrastructure supporting Internet of Things gadgets, data aggregation facilities, and communication channels. After that, it detailed the several types of attacks that may be made against a system, including those that target the device, the network, and the application.

In the second section, we will discuss Trusted Execution Environment and Trusted Platform Module, the two most popular solutions for IoT devices. It detailed the key features, operating principles, and potential flaws in each approach. This demonstrates that although solutions may be effective in certain ways, they are not perfect and need to be refined over time. The final part was a practical for DoS attack prediction using machine learning algorithms.

This thesis shows that options to improve the security of IoT devices do exist and can be obtained quite easily. With any luck, the norm will eventually lead to widespread standardized protections for Internet of Things devices. Protecting private information as the number of connected devices increases must be a need, not a nice-to-have.

REFERENCES

[1] Statista - Internet of Things (IoT) and non-IoT active device connections worldwide from 2010 to 2025,https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/ ,2023.

[2] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: an exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations", IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2702–2733, 2019.

[3] R. Mayer-Sommer, "Smartly analyzing the simplicity and the power of simple power analysis on smartcards", in Cryptographic Hardware and Embedded Systems—CHES 2000: Second International Workshop Worcester, MA, USA, August 17--18, pp. 78–92,2000.

[4] V. Costan and S. Devadas, "Intel SGX explained", Cryptology e Print Archive, 2016.

[5] Y.-S. Won, J.-Y. Park, D.-G. Han, and S. Bhasin, Practical cold boot attack on iot device-case study on raspberry pi", in 2020 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA), pp. 1–4,2020.

[6] J. Deogirikar and A. Vidhate, "Security attacks in IoT: A survey", in 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 32–37,2017.

[7] S. Takarabt et al., "Cache-timing attacks still threaten IoT devices", in Codes, Cryptology and Information Security: Third International Conference, C2SI 2019, Rabat, Morocco, Proceedings-In Honor of Said El Hajji 3 , April 22--24, , pp. 13–30, 2019.

[8] M.-C. Hsueh, T. K. Tsai, and R. K. Iyer, "Fault injection techniques and tools", Computer, vol. 30, no. 4, pp. 75–82, 1997.

[9] S. Ordas, L. Guillaume-Sage, and P. Maurine, "Electromagnetic fault injection: the curse of flip-flops", Journal of Cryptographic Engineering, vol. 7, pp. 183–197, 2017.

[10] C. Bozzato, R. Focardi, and F. Palmarini, "Shaping the glitch: optimizing voltage fault injection attacks", IACR transactions on cryptographic hardware and embedded systems, pp. 199–224, 2019.

[11] H. A. Abdul-Ghani, D. Konstantas, and M. Mahyoub, “A comprehensive IoT attacks survey based on a building-blocked reference model”, International Journal of Advanced Computer Science and Applications, vol. 9, no. 3, 2018.

[12] W. Iqbal, H. Abbas, M. Daneshmand, B. Rauf, and Y. A. Bangash, “An in-depth analysis of IoT security requirements, challenges, and their countermeasures via software-defined security”, IEEE Internet of Things Journal, vol. 7, no. 10, pp. 10250–10276, 2020.

[13] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, “Internet of Things (IoT): Taxonomy of security attacks”, in 2016 3rd international conference on electronic design (ICED), pp. 321–326,2016.

[14] S. Rizvi, A. Kurtz, J. Pfeffer, and M. Rizvi, “Securing the internet of things (IoT): A security taxonomy for IoT”, in 2018 17th IEEE International Conference on Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pp. 163–168,2018.

[15] R. Wang, Z. Jia, and L. Ju, “An entropy-based distributed DDoS detection mechanism in software-defined networking”, in 2015 IEEE Trustcom/BigDataSE/ISPA, vol. 1, pp. 310–317,2015.

[16] Y. Fan, S. Liu, G. Tan, X. Lin, G. Zhao, and J. Bai, “One secure access scheme based on trusted execution environment”, in 2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE), pp. 16–21,2018.

[17] Z. Ahmad, L. Francis, T. Ahmed, C. Lobodzinski, D. Audsin, and P. Jiang, “Enhancing the security of mobile applications by using TEE and (U) SIM”, in 2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing, pp. 575–582,2013.

[18] B. Ngabonziza, D. Martin, A. Bailey, H. Cho, and S. Martin, “Trustzone explained: Architectural features and use cases”, in 2016 IEEE 2nd International Conference on Collaboration and Internet Computing (CIC), pp. 445–451,2016.

[19] S. Pinto and N. Santos, “Demystifying arm trustzone: A comprehensive survey”, ACM computing surveys (CSUR), vol. 51, no. 6, pp. 1–36, 2019.

[20] D. Cerdeira, N. Santos, P. Fonseca, and S. Pinto, “Sok: Understanding the prevailing security vulnerabilities in trustzone-assisted tee systems”, in 2020 IEEE Symposium on Security and Privacy (SP), pp. 1416–1432,2020.

[21] S. Katkin,” Securing sensitive data in modern IoT devices”.

[22] M. Rouf, M. Nazish, I. Sultan, and M. T. Banday, “Implementation of Area and Power Optimised ARM Cortex-M Cores on FPGA”, in 2022 Smart Technologies, Communication and Robotics (STCR), pp. 1–6,2022.

[23] D. Burihabwa, P. Felber, H. Mercier, and V. Schiavoni, “SGX-FS: hardening a file system in user-space with Intel SGX”, in 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), pp. 67–72,2018.

[24] S. P. Rao, G. Limonta, and J. Lindqvist, “Usability and Security of Trusted Platform Module (${$${$${$${$${TPM}$$}$$}$$}$$}) Library {APIs}$”, in Eighteenth Symposium on Usable Privacy and Security (SOUPS 2022), pp. 213–232,2022.

[25] M. Abdel-Basset, M. Mohamed, V. Chang, and F. Smarandache, “IoT and its impact on the electronics market: A powerful decision support system for helping customers in choosing the best product”, Symmetry, vol. 11, no. 5, p. 611, 2019.

[26] A. Abushakra and D. Nikbin, “Extending the UTAUT2 model to understand the entrepreneur acceptance and adopting internet of things (IoT)”, in Knowledge Management in Organizations: 14th International Conference, KMO 2019, Zamora, Spain, July 15--18, 2019, Proceedings 14, pp. 339–347,2019.

[27] S. Adapa, S. M. Fazal-e-Hasan, S. B. Makam, M. M. Azeem, and G. Mortimer, “Examining the antecedents and consequences of perceived shopping value through smart retail technology”, Journal of Retailing and Consumer Services, vol. 52, 2020.

[28] P. Adhi, T. Burns, A. Davis, S. Lal, and B. Mutell, “A transformation in store”, McKinsey & Company, vol. 3, 2019.

[29] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, “A survey on 5G networks for the Internet of Things: Communication technologies and challenges”, IEEE access, vol. 6, pp. 3619–3647, 2017.

[30] A. AlHogail, “Improving IoT technology adoption through improving consumer trust”, Technologies, vol. 6, no. 3, 2018.

[31] A. M. Al-Momani, M. A. Mahmoud, and M. S. Ahmad, “Factors that influence the acceptance of internet of things services by customers of telecommunication companies in Jordan”, Journal of Organizational and End User Computing (JOEUC), vol. 30, no. 4, pp. 51–63, 2018.

[32] S. Madakam, V. Lake, V. Lake, V. Lake, and Others, “Internet of Things (IoT): A literature review”, Journal of Computer and Communications, vol. 3, no. 05, 2015.

[33] W. B. Arfi, I. B. Nasr, T. Khvatova, and Y. B. Zaied, “Understanding acceptance of eHealthcare by IoT natives and IoT immigrants: An integrated model of UTAUT, perceived risk, and financial cost”, Technological Forecasting and Social Change, vol. 163, 2021.

[34] R. P. Bagozzi,” The legacy of the technology acceptance model and a proposal for a paradigm shift”, Journal of the association for information systems, vol. 8, no. 4, p. 3, 2007.

[35] M. S. Balaji and S. K. Roy, “Value co-creation with Internet of things technology in the retail industry”, Journal of Marketing Management, vol. 33, no. 1–2, pp. 7–31, 2017.

[36] S. J. Bell, S. Auh, and A. B. Eisingerich, “Unraveling the customer education paradox: When, and how, should firms educate their customers”, Journal of Service Research, vol. 20, no. 3, pp. 306–321, 2017.

[37] F. Bertacchini, E. Bilotta, and P. Pantano, “Shopping with a robotic companion”, Computers in Human Behavior, vol. 77, pp. 382–395, 2017.

[38] H. N. Boone, D. A. Boone, and Others, “Analyzing likert data”, Journal of extension, vol. 50, no. 2, pp. 1–5, 2012.

[39] S. A. Brown and V. Venkatesh, “Model of adoption of technology in households: A baseline model test and extension incorporating household life cycle”, MIS quarterly, pp. 399–426, 2005.

[40] M. Carcary, G. Maccani, E. Doherty, and G. Conway, “Exploring the determinants of IoT adoption: Findings from a systematic literature review”, in Perspectives in Business Informatics Research: 17th International Conference, BIR 2018, Stockholm, Sweden, September 24-26, 2018, Proceedings 17, pp. 113–125, 2018.

[41] J. Carifio and R. J. Perla, “Ten common misunderstandings, misconceptions, persistent myths and urban legends about Likert scales and Likert response formats and their antidotes”, Journal of social sciences, vol. 3, no. 3, pp. 106–116, 2007.

[42] M. Maksimovic, “Greening the future: Green Internet of Things (G-IoT) as a key technological enabler of sustainable development”, Internet of things and big data analytics toward next-generation intelligence, pp. 283–313, 2018.

[43] S. J. Ceci and W. M. Williams, “Schooling, intelligence, and income”, American psychologist, vol. 52, no. 10, 1997.

[44] A. Chang, “UTAUT and UTAUT 2: A review and agenda for future research”, The Winners, vol. 13, no. 2, pp. 10–114, 2012.

[45] C.-M. Chao, “Factors determining the behavioral intention to use mobile learning: An application and extension of the UTAUT model”, Frontiers in psychology, vol. 10, 2019.

[46] L. Chettri and R. Bera, “A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems”, IEEE Internet of Things Journal, vol. 7, no. 1, pp. 16–32, 2019.

[47] M. Chui, M. Loffler, and R. Roberts, “The internet of things”, McKinsey Global Institute, 2010.

[48] M. Chui, M. Collins, and M. Patel, “The Internet of Things: Catching up to an accelerating opportunity”, McKinsey & Company, New York, 2021.

[49] F. D. Davis, “Perceived usefulness, perceived ease of use, and user acceptance of information technology”, MIS quarterly, pp. 319–340, 1989.

[50] F. D. Davis, R. P. Bagozzi, and P. R. Warshaw, “User acceptance of computer technology: A comparison of two theoretical models”, Management science, vol. 35, no. 8, pp. 982–1003, 1989.

[51] I. J. Deary et al., “Age-associated cognitive decline”, British medical bulletin, vol. 92, no. 1, pp. 135–152, 2009.

[52] M. Dehghani, K. J. Kim, and R. M. Dangelico, “Will smartwatches last? Factors contributing to intention to keep using smart wearable technology”, Telematics and Informatics, vol. 35, no. 2, pp. 480–490, 2018.