A Survey on Privacy Issues in Embedded Systems

Chris V (40232485) || Harleen Kaur (40232489) || Nethra S (40229233)

I. INTRODUCTION

With the continuous and unimpeded rise of the embedded systems market, numerous privacy and security concerns have emerged. If not addressed in a timely and satisfactory manner, these concerns can lead to security vulnerabilities that threaten not only personal information but also critical infrastructure, such as government and military infrastructure.

The objective of our project is to explore and obtain a thorough understanding of the various privacy issues in embedded systems. We explored the reasons behind these privacy concerns, the challenges presently involved, and the major solutions presented by modern-day research. To achieve this objective, we surveyed numerous research journals in this area. Through this report, we offer our detailed and comprehensive analysis of the privacy issues in embedded systems.

The rest of this report is categorized as follows: Section II introduces embedded systems. We explore what embedded systems are and what the different types of embedded systems are. We also look at the generalized architecture of embedded systems to develop a technical understanding of how they work. We also look at the rise of the embedded systems market over the past few decades and discuss its future growth.

Section II explores the existing threats and vulnerabilities in the embedded systems domain. We look at the barriers and challenges involved in securing embedded systems and look at some major attacks against modern-day embedded systems. Following this, we develop an attack taxonomy that helps us classify the different attacks. This process will help us develop signatures for the most common attacks.

In Section III, we analyze some novel and promising solutions presented by modern-day research.

II. BASICS OF EMBEDDED SYSTEMS

A. What are Embedded Systems?

An embedded system consists of hardware and software that are brought together to achieve a very specific purpose. It can be thought of as a very small computer that performs a particular task. Embedded devices sometimes exist as standalone stems, but they are mostly embedded into a larger system, hence the name - embedded systems.

Despite the comparison with computers, an embedded system is in essence nothing like a computer. A computer is a standalone general-purpose device that can perform a wide variety of functions. Our PCs can be used to play games, surf the internet, develop applications and watch movies. On the other hand, an

embedded system has extremely limited functionality. In fact, most often, it has just one function.

B. Types of Embedded Systems

There are numerous types of embedded systems that have a wide range of uses and applications in the real world. In order to develop a general understanding of the importance of embedded systems, it is helpful to classify them based on their functionalities. Embedded systems can be categorized into four major types, depending on their functionalities (1).

Real-Time Embedded Systems: As the name suggests, these are embedded systems that produce real-time results or outputs. The speed at which the result is obtained is a critical performance indicator in real-time embedded systems. Such systems find widespread applications in the defense sector. Some examples of such systems include air traffic controllers, missile defense mechanisms and autonomous vehicle controls. Depending on how important it is to produce real-time results, they can be further classified into soft and hard real-time embedded systems.

Standalone Embedded Systems: As discussed most embedded systems only perform a particular function and are embedded onto a larger system or a host computer. Standalone embedded systems are different in this aspect. They can exist, operate and produce results independently. Digital cameras, watches and calculators are good examples of standalone embedded systems.

Network Embedded Systems: These are embedded devices that rely on internal or external communication networks to generate or relay outputs. Home security systems that generate alerts onto mobile applications remotely are good examples of network embedded systems. They rely on cellular networks as part of their normal functionality. Other examples include banking ATMs which rely network communication with the bank's central databases to obtain the user's account information.

Mobile Embedded Systems: As the name suggests, the key feature that differentiates these embedded systems is portability. It is important to note that mobile embedded systems are different from standalone embedded systems, even if there is some level of overlap between the two. All mobile embedded systems can be considered standalone, but the other way around is not true.

C. The Rise of Embedded Systems

The worldwide embedded systems market touched USD 162.3 billion in 2022 and is expected to grow to more than USD 258 billion by 2032, according to a study conducted by Precedence Research. This represents a compound annual growth rate (CAGR) of 4.77% during the forecast period from 2023 to 2032 (2).

Major sectors and industries that contribute to the unprecedented growth of the embedded systems market include defense, power, healthcare, automotive, automation and consumer electronics. Additionally, the growing popularity of Internet of Things (IoT) devices is also playing a significant role in revenue generation for the embedded systems market. Although all types of embedded systems have gained wide popularity and growth in terms of revenue, standalone embedded systems steals the highest share of revenue. Standalone embedded systems generated 69% of revenue in 2022.

According to the study, the key factors driving growth in this market is the increased use of embedded systems in the healthcare industry, thanks to the deficiencies revealed by the tragic spread of COVID-19 pandemic around the globe. Another driving factor is the increasing popularity of electric and hybrid

vehicles, especially those capable of self-driving.

D. Embedded System Architecture

Evaluating the security and privacy of a system demands an extensive technical understanding of the most intricate workings of the system. Without this knowledge, a comprehensive understanding and evaluation of the privacy issues in embedded systems is not possible. Towards this, we look at the general architecture of embedded systems. This will help understand the weaknesses and vulnerabilities that might open potential attack doors for adversaries. The challenge, however, is that architecture of embedded systems varies depending on the purpose it is built for and the domain it is being applied to.

Most embedded systems consist of a Central Processing Unit (CPU), a Random Access Memory (RAM), a Read-Only Memory (ROM) and several input/output ports. Whether information and data caches are present or not depends on how sophisticated or simple the manufacturer wants the CPU architecture to be. System-wide and local buses are used to facilitate the flow of information within the embedded system [3].

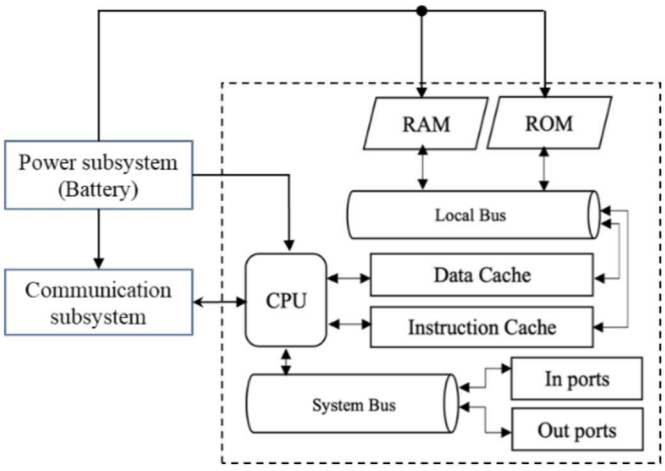

Fig 1. Typical Architecture of an Embedded System [3]

Figure 1 depicts a simple and general architecture of an embedded system. The CPU is clearly the most important and crucial component in an embedded system, but it also requires several other components such as memory and peripheral interfaces. The size and complexity of embedded system architecture is highly variable. It could range from simple electric circuits with small transistors to multi-core microprocessors that support over 1 GHz.

Although this general architecture is useful in understanding how embedded systems work, most modern embedded devices make use of application specific architectures, such FGPA cores and DSPs. Cores optimized for machine learning is also gaining popularity in the industry.

This brings us to System on Chip or SoCs which play a crucial role in modern embedded architecture. In simple terms, an SoC comprises of an integrated circuit which brings together the various building blocks of a computer along with digital and analog interfaces. Depending on the needs of the application, it is possible to build an SoC on a micro-controller or a micro-processor. Embedded systems that require the ability to communicate with the outside world, e.g., network embedded systems, possesses network subsystems. Power subsystems in the embedded system deal with providing and maintaining correct levels of power to the system, many of which are battery powered.

III. Existing Threats & Vulnerabilities

A. Challenges in Embedded Systems Security

In this section, we look at the major challenges involved in designing solutions to privacy concerns in embedded systems.

One of the most significant challenges is the limited processing power available in

embedded systems (4). As a result of this, the use of extensive security solutions such as industry grade anti-malware and advanced intrusion detection systems that are often employed to protect servers and other computers are not feasible in embedded systems. Another related challenge is energy constraints. Embedded systems are mostly designed to work with very less electric power and many work on batteries. Attackers can target this aspect by aiming to drain out batteries leading to failure of these systems (5).

Another challenge is network security. This challenge becomes particularly significant when it comes to Internet of Things (IoTs). Traditional embedded systems were harder to attack as they were designed as isolated islands of technology. This is no longer the case with modern embedded systems which communicate with multiple other devices through local networks and the internet. This exponentially increases the attack surface of embedded systems.

With many manufacturers entering the embedded systems market, more and more designers are forced to emphasize cost cutting measures in order to maximize profits in a crowded market. Even saving a few cents can make a significant difference when it comes to manufacturing millions of units every year.
Some other significant challenges in securing embedded systems include long product lifecycles and difficulties in patching devices. Some embedded systems, like those in vehicles, have an average lifecycle spanning well over a decade or two. As more years pass by, manufacturers have lesser incentives to keep issuing updates and patches, if at all updating is a possibility.

These challenges represent significant barriers to developing secure embedded systems and enabling privacy enhancement solutions.

B. Notable Attacks on Embedded Systems

In this section, we look at some successful attacks against embedded systems. Exploring these attacks will help us gauge the techniques and capabilities of modern-day attackers. Major and significant attacks on embedded systems have been observed since 1982. There has been a major increase in the frequency of attacks since 2001 (6).

The research paper titled "Compromising Industrial Facilities from 40 Miles Away" presents an attack on a key management mechanism used in wireless embedded systems. It demonstrates that some devices are shipped with GUIs that have default values used for device configuration. Upon implementation, the graphical user interface creates a passphrase which will be used to generate an asymmetric encryption key to be used at a later stage. Weaknesses in the srand() function used leads to the attacker being able to calculate the passphrase and encryption key. This enables the attacker to leak communications on the target wireless channel (7).

Several significant attacks against satellite communication systems were presented by the paper titled "SATCOM Terminals: Hacking by Air, Sea, and Land". These attacks on satellite communication systems originate from ground-based sources. One of the attacks presented by the authors involves exploitation of the authentication mechanism on one of the devices. An interface in the SATCOM unit requires an administrator password to access certain critical controls and configurations. Unfortunately, this password used a pattern that is easy to recognize. The password was a combination of the device serial number which was physically printed on the device itself and a hardcoded string. This simple attack enabled attackers to obtain access to restricted administrative controls (8).

A research journal titled "Smart nest thermostat: A smart syp in your home" presents an attack against a popular smart

home automation product, the Nest Thermostat. In this attack scenario, the attacker exploits the fact that the device initiates a reset sequence when a physical button is pressed for 10 minutes. Once the reset sequence is initiated, there is small time period during which the attacker can inject a code through a USB stick connected to the device. This code can be used to boot the device without any cryptographic validation. Such vulnerabilities allow an attacker to install an SSH server enabling the attacker to obtain access to the user's home network (9).

The research paper titled "When firmware modifications attack: A case study of embedded exploitation" discusses an attack on a LaserJet printer through malicious firmware patches. The fundamental reason behind this attack is the use of unauthenticated print commands that the printer accepts and executes. Since the firmware is updated by printing to the memory, an attacker is able to issue a print command to the printer which makes the printer update its firmware with malicious code (10).

IV. Privacy Enhancement in Embedded Systems

A. Towards Cyber-Resilient Embedded Systems

The increasing use of intelligent embedded systems in both industrial and public domains have brought about significant security and privacy concerns that needs to be addressed immediately. The range of applications which can potentially benefit from intelligent embedded systems is growing rapidly. Smart homes, smart cities, smart grids and smart transportation systems are few examples of critical infrastructure that could make use of secure and safe embedded devices.
An intelligent embedded system is one that is able to learn and improve its performance and efficiency. Learning is impossible without the collection of enormous amounts of data to be analyzed and learnt from. This process is highly beneficial in enhancing user experiences, monetizing untapped business opportunities and much more. At the same time, significant risks in relation to data privacy protection inhibit a safe and secure customer journey. Despite this, our use of such technologies has grown exponentially over the years. Considering this, governments and regulatory agencies have been actively enforcing cybersecurity regulations that prevent companies from misusing customer data. These regulations promote the use of cyber resilient devices and systems. Despite extensive efforts by manufacturers, designers and developers to create systems that are secure, most defenses often tend to be passive and ad-hoc.

These shortcomings in security call for a system approach to developing cyber resilient embedded infrastructure. The research paper titled "Establishing Cyber Resilience in Embedded Systems for Securing Next-Generation Critical Infrastructure" (11) presents a systematic approach to improve the resilience of intelligent embedded devices in-line with well-established international security standards.

The authors present following 5 key principles that every embedded system must satisfy in order to be classified "cyber-resilient".

Identify: An embedded system must be able to identify and manage security risks. This can be made possible by conducting an asset management process. It requires a top-down analysis by breaking down the overall system into individual components and evaluating risks and threats to individual components. It is also important to factor in the interaction between the various components and what risks the interaction might create.

Protect: The ability to protect against security attacks is critical. Protecting embedded

systems must consider all the following principles of information assurance.

- Confidentiality prevents unauthorized access to information.
- Integrity prevents unauthorized modification of information.
- Authentication ensures that information can be accessed only by authorized individuals, entities, and processes.
- Availability ensures that information remains available at all times to authorized individuals entities, and processes.

Cryptography-based protection methods can be used to protect embedded systems. Access control methods including ARM TrustZone and Intel SGX are also widely used to isolate subsystems.

Detect: Cyber-resilient embedded systems should be able to effectively detect malicious activity by monitoring behavior. Signature-based and anomaly-based detection systems are popular in the embedded systems industry. Signature-based systems detect attack patterns which have been previously identified. The downside of signature-based systems is that that they work only for known attacks. They are completely ineffective in case of a zero-day attack. Anomaly-based detection systems, on the other hand, defines a range of normal activity and generates alerts if the system's behavior goes outside the defined range.

Respond: The next concept in cyber-resilient embedded systems is a quick and effective response to any attack. Most modern embedded system architectures lack the ability to effectively respond against cyber-attacks, thanks to severe limitations of energy and processing power. Most systems simply implement passive countermeasures like watchdog timers. Such measures generally just resort to rebooting or resetting the system. However, many modern cyber attacks are persistent and cannot be mitigated by reboots. There is a need for improvement in the embedded system microarchitecture to accommodate better attack response mechanisms.

Recover: Last, but definitely not the least, is the ability to recover the device back to a healthy state after attack mitigation. This involves repairing the system to bring It back to the pre-attack performance level and issuing updates to patch vulnerabilities that allowed the attack to take place. This is another area which is lacking in most modern embedded system architectures. They are mostly focused on the principles of reliability to perform recovery. This approach proves inefficient in the development of cyber-resilient systems.

Based on these principles and requirements, the authors propose three core microarchitectural characteristics that allows the establishment of historical data stream through continuous monitoring of resources and actions. These characteristics are:

Independent Active Runtime System Security Manager: This system is responsible for protection, detection, response and recovery. It should possess the ability to continuously monitor the usage of resources and make use of this information to detect anomalies which could potentially lead to malicious attacks. It should also be able to respond to successful attacks and recover back to a healthy state post attack mitigation. A critical feature is that this security manager should exist in a containment region, isolated from other components of the system.

Active Runtime Resource Monitor: The primary function of these components, as the name suggests, is monitor specific behavior to detect malicious activity. Upon detection of malicious/abnormal activity, it is reported to the security manager. As embedded system architectures become more complex and sophisticated, resource monitors become essential in the secure function of embedded systems. In addition to attack detection, the

collected data also play a major role in investigation of the breach and defining signatures to prevent similar attacks in the future.

Active Response Manager: The role of the response manager is to implement response and recovery mechanisms in the event of a successful breach. It should enforce response and recovery strategies as defined by relevant security policies.

Through this paper, the authors develop a systematic and comprehensive approach to developing cyber-resilient intelligent embedded systems. The most important principles and requirements for security systems were discussed. The authors also proposed three core microarchitectural characteristics in order to resolve existing deficiencies in modern embedded system architectures.

B. Privacy Enhancement in Embedded Smart Cameras

In this section, we explore some techniques to enhance privacy of individuals who are being recorded using embedded smart cameras. Smart camera networks are increasingly popular among governments and law enforcement agencies in their efforts to prevent criminal and unlawful activities. These techniques allow law enforcement agencies not only to prevent crimes, but also to conduct investigations and uphold accountability.

Other than applications in traditional surveillance, smart camera systems are also useful in other sectors including elderly, home security and entertainment. The increasing popularity of smart cameras has resulted in them being popular targets for adversary attacks. Such attacks compromise the right to privacy of citizens and individuals who are being recorded using these devices. For instance, the use of cameras to prevent crime is incredibly useful in law enforcement, but it is important to note that these cameras are constantly recording even in the absence of crime or criminals. Most subjects who are being recorded by these cameras are civilians whose right to privacy might be violated because of these surveillance cameras.

The research paper titled "TrustCAM: Security and Privacy-Protection for an Embedded Smart Camera based on Trusted Computing" (12), presents a solution to this very problem. The authors envision to solve this problem through the use of Trusted Computing. A Trusted Platform Module (TPM) is used for this purpose. The implementation process of TrustCAM involves two key processes: (i) Image Encryption and (ii) Image Signing.

The authors propose the use of encryption to protect the sensitive regions of the image, including subjects of surveillance cameras. In the proposed model, encryption is performed using two 256-bit AES session keys.

Figure 2 depicts the process flow involved in encryption of sensitive regions of the image. The image is passed onto the privacy protection system by the image acquisition system which in turn obtains them from the sensors. Following are the major steps and actions performed by the privacy protection system to encrypt sensitive regions:

Firstly, the regions of interest are identified. These regions of interest are the regions that are going to be encrypted to protect privacy. For example, the face of pedestrian recorded by surveillance camera could be the region of interest. Once the ROI is identified, they are extracted from the input image.

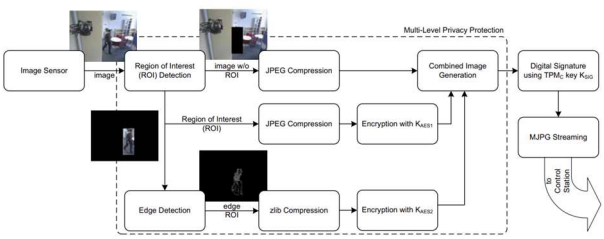

Fig 2. Process Flow of Image Encryption in TrustCAM(12)

Following the extraction process, the remaining background image along with the region of interest are compressed using libjpeg. Then, canny edge detection is performed on the ROI image. The result of this process is a black and white image which allows us to observe the actions of subjects in the video without revealing their identity. This image is then compressed using zlib, since it is more efficient for compressing binary images. The compressed ROI images are then encrypted with the 256-bit AES session keys.

Once the encryption process is complete, the authors then move to signing the image. For this, the SHA1 hash sum of image is first computed. This takes around 2 ms. Following the computation, hash value is digitally signed inside the TPM. Due to prohibitive time constraints, the authors implement digital signature on sequences of images instead of every individual image. The computation of signature is time-consuming process. Since the TPM commands are executed in parallel to the processor, the system is able to continue with accumulating the hash sum of the next group of images without waiting for the result of current signature computations. The signatures are simply attached to the frames once the TPM completes the computations.

Through these techniques, the authors are able implement a secure and privacy-enhanced smart camera platform that preserves the privacy of recorded individuals and encrypts sensitive image regions using cryptographic keys. Experiments and analysis performed by the authors reveal that the additional computation overhead is minimal and non-prohibitive.

C. Exploring & Resolving Privacy Leaks in Smart Embedded Wearable Devices

Wireless embedded sensing technologies are gaining wide popularity among all audiences. The range of applications presented by wearable sensors is enormous, especially in the wake of aeging populations and the rise of chronic illnesses. Fitibit, Jawbone UP and Nike+ Fuelband SE are some popular wearable activity monitors in the market. However, a key disadvantage with these activity monitors is inability interoperate with other sensing technologies and the limited availability of raw data. In response to these limitations, wearable sensors are gaining more and more popularity along with other sensing technologies to form infrastructures with multiple nodes. These technologies are capable of co-relating how daily activities affect well-being.

The hallmark of such technologies is personalized suggestions and recommendations that are based on the wide range of personal data collected. Despite the enormous benefits offered by these technologies, privacy concerns regarding disclosure and misuse of personal data is major inhibitor to widespread adoption of such devices. The paper titled "Privacy Leakage of Physical Activity Levels in Wireless Embedded Wearable Systems" (13) explores some of these privacy concerns. They experimentally demonstrate that it is possible to leak physical activity levels of users wirelessly from embedded wearable devices.

In the experimental demonstration, Alice, the victim, uses a wearable sensor to monitor physical activity. The sensor consists of several components including a triaxial accelerometer which senses all movements and vibrations made in three orthogonal directions, and a wireless radio which transmits data to other nodes in the system. The Integral Modulus of Acceleration, which is commonly used for measuring and estimating physical activity levels, transforms the raw acceleration data into activity levels. Needless to say, it is assumed that data concerning physical activity levels are private to Alice. The attacker, Eve, does not have any physical access to the sensor. However, Even is assumed to be close enough to Alice to be able to receive wireless radio signals transmitted by the device.

During the attack, Eve continuously listens to all encrypted packets of data transmitted by the device through its wireless channels. The packets contain raw acceleration data. Eve collects these wirelessly transmitted packets and calculates the power of each received signal, termed Received Signal Strength (RSS). The Integral Modulus of Acceleration (IMA) co-relates to the first derivative of the Received Signal Strength. The dynamic levels of human activity makes this co-relation inevitable. At this point, it is useful to define a successful attack.

The authors define it a significant co-relation between the encrypted signal and the rolling standard deviation calculated by Eve. The authors validate their experiment with data received from a user living in a prototype smart home. The first two plots in the figure show Alice's encrypted signal. The first is the magnitude of acceleration and the second is the IMA for a window of 500 samples. The remaining three plots represent signals obtained by Eve. These include the raw RSS, the first derivative of RSS and the rolling standard deviation of the derivative. The authors demonstrate from the values calculated by Eve that there is a strong correlation between the IMA and rolling standard deviation. As a result, the authors conclude that the privacy attack is successful as defined.

The authors also present a methodology to prevent such attacks in order to enhance privacy of these wearable systems. The attack is made possible primarily due to the direct correlation between the physical activity levels of Alice and the strength of signals received by Eve. As a result, the attack can be prevented through the introduction of random artificial variations in transmission powers generated by Alice's device. These prevents Eve from being able to deduce any meaningful inferences from the Received Signal Strength, rendering the entire attack impossible.

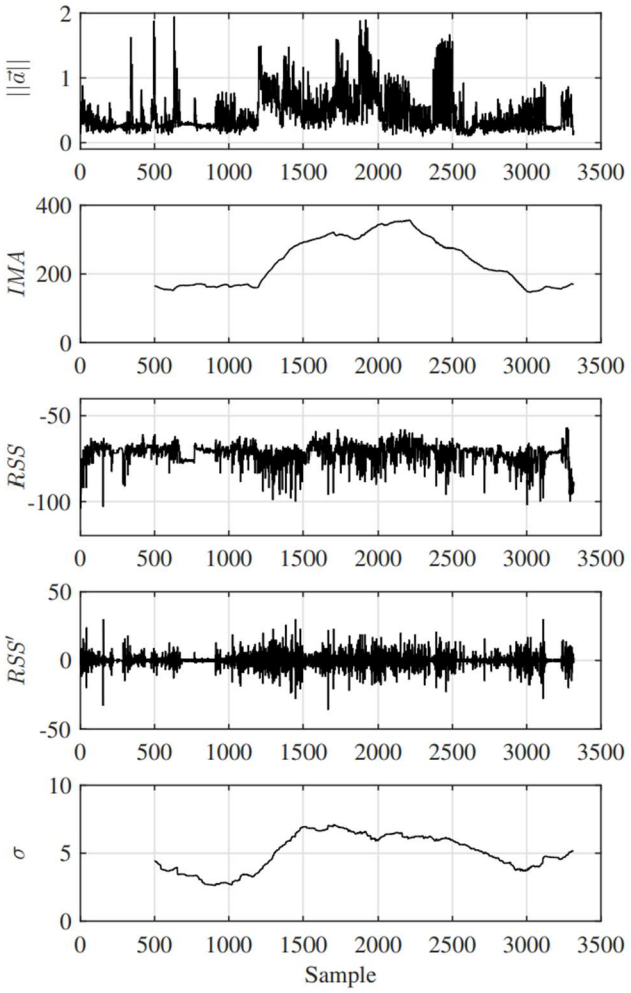

Fig 3. Experimental results of demonstrative attack [13]

D. Privacy and Security issues in Smart Home Architecture

The world of smart homes has changed dramatically in the last few years, from basic remote on/off switches to complex settings where Internet-connected gadgets control appliances and systems in the home. The Internet of Things (IoT), which includes devices with sensors, actuators, software, and data processing capabilities, is what is driving this evolution. Perception, transport, network, and application layers make up the complex layered architecture of IoT Smart Homes. These layers cooperate to offer seamless services, access, connection, and management of IoT devices via the Internet from any place. The

physical elements of the system, such as sensors and actuators, are the focus of the perception layer, which records external data like temperature, motion, light, and door status. This data is transmitted over the network more easily by the transport layer than by the network layer.

Device management is done by the application layer via dashboards or remote monitoring. Data transmission is the process of sending data wirelessly across LPWAN from endpoints to a cloud platform using protocols like HTTP and MQTT. The collected data is stored in cloud storage, where it is later processed and examined using cloud engines. Through online or mobile applications, the user interface enables the control and visualization of the smart home environment. Voice commands are becoming more and more common thanks to Voice User Interfaces (VUI) like Alexa and Google Home. The gateway device ensures secure and filtered communication by bridging the gap between end devices, sensors, systems, and the cloud. This all-inclusive architecture prioritizes security, connectivity, and efficiency in the quickly developing field of smart homes.

The rapid advancement of technology and innovation has encouraged people to embrace a more convenient lifestyle enabled by smart home appliances. These gadgets have many benefits, especially in terms of safety and security, but they are not impervious to cybersecurity threats.

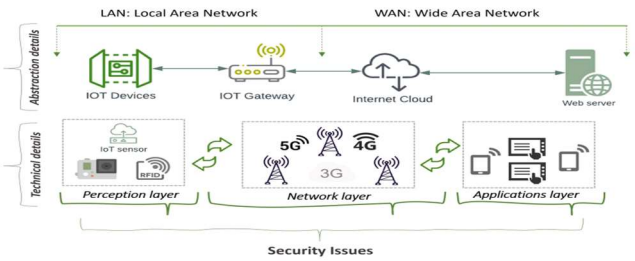

Fig. 4. Architecture of IoT system

There have been documented cases of intelligent home automation being compromised, which underscores the possible weaknesses that result from individuals' carelessness and gadget vulnerabilities. For those who rely extensively on smart home automation, this cybersecurity risk is a serious problem that calls for increased knowledge of safety and security precautions. An enhanced review of network-based processes, a more detailed assessment of the cybersecurity framework, and an analysis of threat areas are necessary to address these issues. Smart home systems are vulnerable because they lack sufficient built-in security mechanisms and unclear user instructions.

The revolutionary effect that smart homes have on their customers' life, emphasizing the utilization of several sensors such as activity loggers, motion detectors, cameras, and microphones. Although voice-activated lighting and remote-controlled door locks are practical aspects of smart homes, security experts have found substantial hazards related to privacy and security. These vulnerabilities are highlighted in the narrative. Concerns include devices with vulnerabilities that might be used for remote spying or interfering with inhabitants' life, as well as unsecured communication that could result in the disclosure of personal information.

A source (reference [17]) is mentioned, which perhaps offers further information on these privacy and security concerns related to smart homes. Although smart home systems aim to improve people's lives by providing helpful services, there are significant privacy concerns due to the ongoing problem of data leaks.

The Transport layer, Perception layer, Application layer, and Network layer are the four interrelated levels of connected automation systems that must be considered while building a smart home. The functionality and complexity of smart home installations are influenced by these levels taken together.

Layer	IoT device/Application	Purpose	Attack object
Perception	Physical objects, Sensors, Actuators	Collect information from sensors/devices	Physical damage, Eavesdropping, Node capture, Replay attack, Timing attack
Network/Transport	Router, Gateways, LoraWAN, 3G, 4G	Connect devices to each other and higher layer through wired/wireless media	Full control, Eavesdropping, Traffic analysis, DoS Attack, Man in the middle, DDoS
Application	Household appliances	Has the responsibility to extend sensor-specific service to application/clients	Take control, To identify speakers, Cross-site scripting, Malicious canr overflow

Table1. Smart Home Attacks at Different Levels of Layers

Table 1 shows how IoT devices are distributed throughout each tier and describes the various kinds of attacks related to smart home systems.

Transport Layer:

Specifically designed for IoT and M2M message delivery, MQTT (Message Queuing Telemetry Transport) uses TCP for data transport. The entire communication is fully encrypted thanks to IP/TCP and Transport Layer Security. This protocol is a favorite option in many Internet of Things applications, especially those with limited environments, because it is especially useful for managing event-based or streaming data. Its effectiveness in transmitting payload adds even more credence to its appeal.

Contrarily, CoAP (Constrained Application Protocol) depends on UDP security mechanisms to protect data. CoAP chooses Datagram TLS over UDP, whereas HTTP uses TLS over TCP, offering security that is adapted to the limitations of IoT devices. It is made with lightweight communication in mind, catering to the special needs of confined devices in smart homes.

Replay Attack: Replay attacks include a third party listening in on conversations between reliable parties and copying and storing a valid service request. The attacker then plays it again in order to obtain unauthorised access. The integrity of smart home services is threatened by this, since it replays saved, valid requests, permitting unauthorized use.

Message Modification: Attackers may try to change communications by listening in on talks between people who are authorised to communicate. This could entail altering data values or maliciously updating software. The integrity and dependability of information shared inside the smart home network are jeopardized by message alteration.

Denial of Service (DOS) Attack: When a hacker wants to prevent authorized users from accessing the network or limit the availability of network services, they utilise denial-of-service (DoS) assaults. The attacker exhausts resources by sending an unending stream of messages into the smart home network. These kinds of attacks interfere with the availability of network services, limiting authorized users' access and interfering with internal traffic in smart homes.

Perception Layer:

The perception layer of a smart home functions similarly to the eyes, hearing, and nose of a human, and is frequently compared to these senses in humans. This layer's job is to recognise items in the surrounding environment and get pertinent data. A variety of sensors, including MEMS (Micro-Electro-Mechanical Systems), RFID, and 2-D barcode scanners, are carefully chosen according to applications' needs.

Sensor Functionality:

One important sensor in this layer, MEMS, monitors the physical and environmental conditions inside dwellings by detecting changes in the surroundings. Gyroscopes also measure rotational motion, which enables the system to recognize events such as the opening and closing of doors or windows. Position sensors can identify people or things in a specific area thanks to this degree of motion tracking.

Home Security Applications:

These sensors gather information on location, air quality, environment, motion, and vibration, among other important topics. To improve home security, homeowners can remotely

monitor their windows, doors, and appliances. However, attackers looking to modify or breach these sensors find that this abundance of information makes them a tempting target.

Eavesdropping: Eavesdropping is an attack in which a perpetrator listens in on private conversations with the intention of stealing information sent across a network. Data sent and received can be accessed by attackers by taking advantage of insecure communication.
The Dolphin assault is noteworthy because it involved hackers trying to control mobile devices by concealing voice commands on ultrasonic carriers. Vulnerabilities were examined in well-known speech recognition programs like Siri, Google Now, and others.

Node Capture: By using this approach, a hacker can gain access to the entire network and expose sensitive security data including cryptographic keys, shared secrets, and sender-receiver interactions.

Replay Attack: Replay attacks, often called interceptor attacks, entail a hacker listening to a conversation between a sender and a recipient to steal real data. The recipient unwittingly fulfils the intruder's purpose is caused by the intruder's presentation of this information to the victim, which is coded to look like a legitimate request.

Timing Attacks: Timing attacks, which are frequently used against devices with low processing power, include timing how long a system takes to react to different input, requests, or cryptographic techniques. Attackers seek to locate vulnerabilities and pilfer confidential information.
Application Layer:

Activating the Smart Home's Features: In a smart home setting, the application layer is essential for controlling how IoT devices and the network communicate. This layer offers a variety of applications that make it easier to monitor and manage different devices in a smart home. On the other hand, application-level vulnerabilities can seriously jeopardize the security of the Internet of Things network, jeopardizing sensitive personal information.In the heterogeneous Internet of Things environment, application layer protocols play a critical role in establishing the complex connections between devices and the network. The efficiency of a smart home ecosystem depends on the smooth operation of its applications.

XSS (Cross-Site Scripting): An example of an injection assault is XSS, which is a common online attack. It is not hard to identify, yet it presents problems for defense and differentiation. It includes distributing malicious web code, frequently in script form, through specific web applications and usually targets user-side applications. This may result in the loss of cookies or personal data, giving hackers access to user sessions and even whole machines.

Malicious Code Attack: To damage the system, malicious code attacks entail inserting software code. To steal user data, attackers may employ end-user vulnerabilities to introduce several kinds of malicious code. Notably, traditional anti-virus software may not be able to successfully block or manage this kind of danger.

Buffer Overflow: Buffer overflow attacks are just one of the many software and hardware flaws that can affect Internet of Things devices. In this kind of assault, the attack happens when the storage space is surpassed, and a buffer acts as a temporary storage area for data. A hardware design with architectural improvements for identifying buffer overflow attacks is presented by the authors in (14).

Networking Layer:

Numerous network layer protocols, such as Bluetooth, IrDA, WiFi, ZigBee, RFID, NUWB, NFC, and Wireless Hart, are necessary for smart home devices. These methods of communication create links between servers, networks, and devices. Nonetheless, attackers frequently target the network layer, with a

special emphasis on wireless networks. Inadequate confidentiality settings and authentication present major vulnerabilities in network communication, giving bad actors opportunities to take advantage of weak networks. Insufficient confidentiality settings and authentication processes are the main causes of concern when it comes to network connectivity in smart homes. Attackers may use these flaws to undermine the system's overall security.

Denial of Service (DoS) Attack: By flooding devices or network resources with requests, a denial-of-service (DoS) attack seeks to prevent legitimate users from accessing those devices or resources, making them difficult or impossible to use.

Man-in-the-Middle (MiTM) Attack: A Man-in-the-Middle (MitM) attack is a severe threat to internet security because the attacker surreptitiously listens in on the sender and recipient's conversation and modifies it. A Samsung smart refrigerator bug that allowed unauthorized access to network data is one example from recent times.

DDoS (Distributed Denial of Service): DDoS assaults are a serious risk to the internet. Attackers frequently take advantage of TCP/IP protocol flaws by flooding the victim with packets through reflection and amplification techniques. Among the methods are TCP Syn Flood, UDP Flood, and ICMP Flood.

E. Security Precautions to Prevent Attacks:

Cybersecurity risks are introduced when typically, isolated smart devices—like locks, appliances, and lighting—are integrated into networked smart homes. This could result in threats and assaults. Examples of cybercriminals interacting via hacked baby monitors highlight how susceptible smart home appliances are. It is advised to take the following safety measures to reduce these risks:

Router Security Measures:

Making sure routers are configured correctly is a crucial first step in protecting smart homes. This entails updating the router and putting security measures in place, such changing the password and eliminating default names. It also entails using strong encryption standards like WPA3.

AI-Rendered Home Defense:

A paradigm shift is brought about by the incorporation of Artificial Intelligence (AI) into home security, which enables real-time data access and facial recognition without the need for the internet. Artificial intelligence (AI)-enabled gadgets are essential for threat analysis, facial recognition, and security process automation in smart homes.

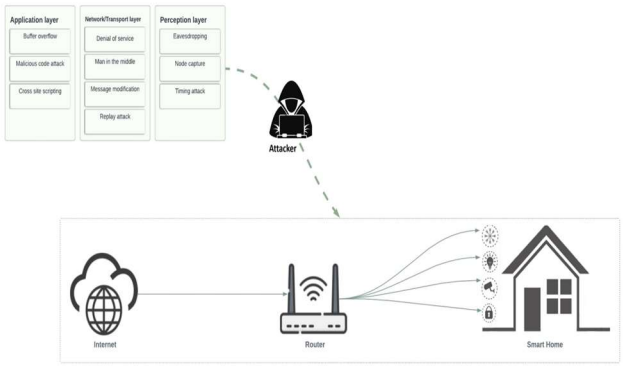

Fig.5. Security attacks in smart homes

Motion Security using Deep Learning:
Motion detection and recognition are the main uses of deep learning in smart homes. Human motion patterns are recognized using Convolutional Neural Network (CNN) models, which achieve excellent accuracy in classifying motions as belonging to either occupants or possible intruders.

Strategies for Managing Passwords:

Password strength is still a critical component of cybersecurity for smart homes. Creating

strong, one-of-a-kind passwords for a variety of accounts—such as Wi-Fi networks and IoT device logins—represents best practices. One useful tool for efficient and safe password management is password manager.

Implementation of Convolutional Neural Networks (CNNs):

Motion recognition in smart home security systems is improved by using CNN models. Areas of interest are found in surveillance camera images by processing, which enables effective motion detection classification and prompt user notifications.

IoT-Dedicated Wi-Fi Networks:

It is advised to create a dedicated Wi-Fi network for IoT devices to improve security. This separation strengthens overall cybersecurity by preventing IoT devices from potentially accessing important parts of the home network.

Feature Management to Improve Security:

One of the most important things you can do to reduce potential vulnerabilities on IoT devices is to disable superfluous functionalities. Users can drastically lower the number of possible entry points for cyber-attacks by turning off remote access functions when they're not needed.

Protections for Remote Third-Party Services:

User privacy must be prioritized in smart home systems that integrate remote services. Users connecting with third-party services can be assured of enhanced security and privacy with the implementation of customizable time-resolution restrictions and thorough monitoring of API visits.

Updates to Firmware and Security Protocols:

To fix security flaws, Wi-Fi routers and Internet of Things (IoT) devices must have their firmware updated on a regular basis. The protocols known as Secure Firmware Over-The-Air (FOTA) are essential for guaranteeing the safe transmission and implementation of upgrades.

Adoption of Multi-Factor Authentication:

It becomes clear that multi-factor authentication is a strong security precaution, requiring more identity verification levels than just passwords. Overall security is greatly improved by enabling this feature, either through third-party authenticators or device-specific implementations.

Safe Communication between Machines and Machines (M2M):

Enforcing safe communication protocols is essential for machine-to-machine communication in smart homes. The integrity of M2M communication is strengthened by confidentiality protections combined with lightweight authentication systems.

Deployment of Next-Generation Firewalls:

Advanced security capabilities like virus protection, content filtering, and intrusion detection are introduced with the deployment of NGFWs. Investigating FPGA-accelerated architectures and Software-Defined Networking (SDN) substantially improves smart home network threat detection capabilities.

F. Physical Security Issues in Embedded Systems

Given that deployed devices are frequently left unattended, especially in hostile contexts, the physical layer of embedded systems (ESs) raises important questions. This section explores facets of physical security in ESs and emphasizes the dangers of interfering with devices.

Side Channel Attacks:

The possibility of side channel attacks (SCA) is a major factor in physical security. A malevolent entity can launch a variety of attacks, including micro-probing, reverse engineering, and complex side-channel attacks, if it manages to physically access a non-tamper-resistant device. Timing assaults, simple power analysis (SPA), differential power analysis (DPA), and their electromagnetic cousins, SEMA and DEMA, are a few examples of side-channel attacks. Furthermore, assaults known as differential faults (DFA) present a potential threat.

Protection of Power Supply:

Power Management in Embedded Systems:
A steady and continuous power supply is necessary to guarantee the correct operation of the electronic components in embedded devices. This entails keeping current and voltage levels within predetermined bounds. The power supply should also be able to keep an eye on its own condition and react suitably to any problems that are identified that would compromise the system's capacity to function normally. It becomes essential to implement fail-safe methods in hardware, software, or both to safeguard the device and stop potential damage from propagating throughout its components.

Cryptographic Mechanisms:

Specifically designed algorithms and implementations for devices with limited resources are needed, and this is what Lightweight Cryptography (LWC) aims to solve. These gadgets include mobile devices, contactless smart cards, sensor nodes, and RFIDs. LWC's main goal is to offer cryptographic solutions that strike a compromise between the demands of critical security and the constraints placed on these devices' limited resources.

The major initiatives in LWC are to optimize cryptographic algorithms for use in situations where excessive energy consumption, low processor power, or memory constraints make classic algorithms impracticable. Keeping security on par with conventional cryptography techniques while reducing the difficulties brought on by the "battery gap" is the major objective. The significant energy consumption overheads associated with maintaining security features on devices with constrained battery capacity are called the "battery gap."

G. Open research issues in the context of security provision of CPS:

Facilitating Cooperative Mechanisms for CPS Defense:

In CPS protection, collaborative procedures entail a common approach to threat mitigation and information sharing regarding threats faced. It is believed that traditional dispersed security techniques are useless against contemporary threats. The McAfee analysis highlights the necessity of an open, integrated ecosystem that enables security system cooperation, even when the systems are owned by various parties. By facilitating the sharing of threat intelligence, this cooperative strategy lowers the need for resources and response times. Multiple parties improving upon a common knowledge base increases the effectiveness of threat mitigation initiatives.

Safety Information Sharing for Dispersed Locations:

One of the biggest challenges in geographically dispersed CPS components is ensuring secure communication. Dispersed key nodes or subsystems might make communication more difficult. The risk of compromised applications gaining access to sensor node data highlights the need for robust security measures. Inspired by federated learning, one viable option is to process the data locally instead of sending it to a central node. With this method, updating a

global model takes precedence over sending local datasets to a central server.

safeguarding vital infrastructure

Smart grids provide an example of the issues that critical infrastructure faces. These include heterogeneity in technology, vulnerabilities in communication protocols, physical device limits, and a variety of security techniques. To tackle these obstacles, all-encompassing security measures are needed, such as effective communication protocols and methods to guarantee the integrity of physical objects.

Safeguarding Digital and Physical Elements:

Integrity is a basic prerequisite for CPS, which applies to both overarching systems and sensor networks. The use of proprietary solutions with unsafe techniques results from the lack of a standard methodology for creating secure CPS.

Mobility and Security:

Because mobile devices frequently interface with external networks, they present security vulnerabilities. Examples of these technologies include wearable smart assistants and implants. Given the possible risks to human health and life, it is imperative that these devices be secured. To reduce these threats, mobile devices must have improved security features included.

Proactive Security Mechanisms:

Analytical tools and proactive security systems are essential for spotting and mitigating such risks. Based on past data, analytical tools can recommend main attack pathways and threat categories. Improving system security requires that security solutions be tailored to certain threat types, including DDoS attacks.

Tools for Integration and Analysis:

It is necessary to develop new security models that include human-assisted analysis, decision-making algorithms, and machine learning. Analysis with human assistance is still necessary to discover previously encountered threats. Due to the large amounts of data involved, it is critical to find solutions for faster response times as real-time responses to the most frequent threats become necessary.

V. Conclusion

As the embedded systems market continue to grow, their prevalence in critical infrastructure and our personal spaces have become a crucial motivator for a growing number of privacy attacks on embedded systems. As a result, developing efficient and comprehensive privacy solutions that protect embedded systems is extremely important. However, there are multiple challenges to be dealt with when developing security solutions for embedded systems. Unlike desktop computers, embedded systems have very less processing power and energy constraints. Due to these challenges, conventional security mechanisms are not feasible. Targeted security solutions need to be developed in order to address these concerns. Many researches offer promising solutions. Through this project, we examined some of the most promising embedded systems privacy solutions at the forefront of modern research. We analyze their relevance, feasibility, and shortcomings. Thanks to this survey, we were able to achieve a comprehensive understanding of the challenges in this domain and many innovative techniques that are still being developed.

VI. References

(1) Brett Daniel, July 2022, "What Are Embedded Systems?", Blogs by Trenton Systems. https://www.trentonsystems.com/blog/what-are-embedded-systems

(2) Presedence Research, 2022, "Embedded Systems Market". https://www.precedenceresearch.com/embedded-

systems-
market#:~:text=The%20global%20embedded%20systems%20market,period%20from%202023%20to%202032.&text=Key%20Takeaways%3A,51%25%20revenue%20share%20in%202022.

[3] Aloseel, Abdulmohsan, Hongmei He, Carl Shaw and Muhammad Khan. "Analytical Review of Cybersecurity for Embedded Systems." IEEE Access 9 (2021): 961-982.

[4] S. Ravi, P. Kocher, R. Lee, G. McGraw, and A. Raghunathan, "Security as a new dimension in embedded system design," in Proceedings of the 41st annual conference on Design automation -DAC '04, 2004, p. 753, doi: 10.1145/996566.996771.

[5] C. Bodei, S. Chessa, and L. Galletta, "Measuring security in IoT communications," Theor. Comput. Sci., vol. 764, pp. 100–124, 2019, doi: 10.1016/j.tcs.2018.12.002.

[6] M. Keefe, "Timeline: Critical infrastructure attacks
increase steadily in past decade," 2012. [Online].

[7] L. Apa and C. M. Penagos, "Compromising Industrial Facilities from 40 Miles Away ," ser. BlackHat, 2013.

[8] R. Santamarta, SATCOM Terminals: Hacking by Air,
Sea, and Land, IOActive, Inc., 2014. [Online). Available:
https://www.defcon.org/images/defcon-22/dc-22presentations/Cerrudo/DEFCON-22-Cesar-Cerrudo-Hacking-Traffic-Control-Systems-UPDATED.pdf

[9] G. Hernandez, O. Arias, D. Buentello, and Y. Jin, "Smart nest thermostat: A smart syp in your home," ser. Black Hat, 2014

[10] A. Cui, M. Costello, and S. J. Stolfo, "When firmware modifications attack: A case study of embedded exploitation," in Proceedings of NDSS Symposium 2013, 2013.

[11] F. Siddiqui, M. Hagan and S. Sezer, "Establishing Cyber Resilience in Embedded Systems for Securing Next Generation Critical Infrastructure," 2019 32nd IEEE International System-on-Chip Conference (SOCC), Singapore, 2019, pp. 218-223, doi: 10.1109/SOCC46988.2019.1570548325.

[12] T. Winkler and B. Rinner, "TrustCAM: Security and Privacy-Protection for an Embedded Smart Camera Based on Trusted Computing," 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance, Boston, MA, USA, 2010, pp. 593-600, doi: 10.1109/AVSS.2010.38.

[13] X. Fafoutis, L. Marchegiani, G. Z. Papadopoulos, R. Piechocki, T. Tryfonas and G. Oikonomou, "Privacy Leakage of Physical Activity Levels in Wireless Embedded Wearable Systems," in IEEE Signal Processing Letters, vol. 24, no. 2, pp. 136-140, Feb. 2017, doi: 10.1109/LSP.2016.2642300.

[14] P. Mann, N. Tyagi, S. Gautam, and A. Rana, "Classification of various types of attacks in IoT environment," in Proc. 12th Int. Conf. Comput. Intell. Commun. Netw., 2020, pp. 346–350.

[15] B. Xu et al., "A security design for the detecting of buffer overflow attacks in IoT device," IEEE Access, vol. 6, pp. 72862–72869, 2018.

[16] E. Džaferovic, A. Sokol, A. A. Almisreb, and S. M. Norzeli, "Dos and ´ DDoS vulnerability of IoT: A review," Sustain. Eng. Innov., vol. 1, no. 1, pp. 43–48, 2019.

[17] E. Zeng, S. Mare, and F. Roesner, "End user security and privacy concerns with smart homes," in Proc. 13th Symp. Usable Privacy Secur., 2017, pp. 65–80. [Online). Available: https://www.usenix.org/conference/soups2017/technical
sessions/presentation/zeng

[18] Aldahmani, Aaesha, et al. "Cyber-security of embedded IoTs in smart homes: challenges, requirements, countermeasures, and trends." IEEE Open Journal of Vehicular Technology 4 (2023): 281-292.

[19]FysarakisK.,HatzivasilisG.,RantosK.,PapanikolaouA. and ManifavasC, "Embedded Systems Security Challenges."
InternationalConferenceonPervasiveandEmbeddedComputingandCommunicationSystems (MeSeCCS-2014), pages 255-266