Securing the Internet of Things: Challenges, Threats and Solutions

Panagiotis I. Radoglou Grammatikis[a], Panagiotis G. Sarigiannidis[a,*], Ioannis D. Moscholios[b]

[a]*Department of Informatics and Telecommunication Engineering, University of Western Macedonia, Kozani, Greece*
[b]*Department Informatics and Telecommunications, University of Peloponnese, Tripolis Greece*

Abstract

The Internet of Things (IoT) is the next technological leap that will introduce significant improvements to various aspects of the human environment, such as health, commerce, and transport. However, despite the fact that it may bring beneficial economic and social changes, the security and the privacy protection of objects and users remain a crucial challenge that has to be addressed. Specifically, now the security measures have to monitor and control the actions both of users and objects. However, the interconnected and independent nature of objects, as well as their constrained capabilities regarding the computing resources make impossible the applicability of the conventional security mechanisms. Moreover, the heterogeneity of various technologies which the IoT combines increases the complexity of the security processes, since each technology is characterized by different vulnerabilities. Furthermore, the tremendous amounts of data which is generated by the multiple interactions between the users and objects or among the objects make harder their management and the functionality of the access control systems. In this context, this paper intends to provide a comprehensive security analysis of the IoT, by examining and assessing the potential threats and countermeasures. More detailed, after studying and determining the security requirements in the context of the IoT, we implement a qualitative and quantitative risk analysis, investigating the security threats per layer. Subsequently, based on this process we identify the suitable countermeasures and their limitations, paying special attention to the IoT protocols. Finally, we provide research directions for future work.

Keywords: Countermeasures, Cyberattacks, Internet of Things, Privacy, Protocols, Risk Assessment, Security

1. Introduction

The IoT represents a technologically optimistic future, where the objects will be able to utilize the Internet and make intelligent collaborations with each other anywhere and

*Corresponding Author
Email addresses: pradoglou@uowm.gr (Panagiotis I. Radoglou Grammatikis), psarigiannidis@uowm.gr (Panagiotis G. Sarigiannidis), idm@uop.gr (Ioannis D. Moscholios)

This work appears in Elsevier Internet of Things, Volume 5, pp.41-70, 2019, ISSN: 2542-6605, DOI: https://doi.org/10.1016/j.iot.2018.11.003, URL: "http://www.sciencedirect.com/science/article/pii/S2542660518301161

anytime. In particular, the IoT combines a wide range of technologies, such as sensors, actuators, Internet, cloud computing as well as many communication infrastructures. While the term of IoT was coined in 1999 by Ashton [1], the idea of this technology was envisioned many years ago. In more detail, Nikola Tesla in his interview in the Colliers magazine, in 1926, said that: "When wireless is perfectly applied the whole earth will be converted into a huge brain, which in fact it is, all things being particles of a real and rhythmic whole and the instruments through which we shall be able to do this will be amazingly simple compared with our present telephone. A man will have the ability to carry one in his vest pocket" [2, 3]. In 1950, the British scientist Alan Turing quoted that: "It can also be maintained that it is best to provide the machine with the best sense organs that money can buy, and then teach it to understand and speak English. This process could follow the normal teaching of a child" [4, 5].

Today, this kind of technology is found in many application fields such as, energy industry, health and transportations [3]. According to Gartner, over than 28 billion IoT devices will have the ability to connect to the Internet by 2020 [6]. It is estimated that the number of the human world community will approach 7.8 billion by 2020; therefore, as a result, each human will possess on average three devices which will be able to connect to the Internet. [7]. There are many standardization organizations both from academia and industry which have defined the IoT term. In this paper, we suggest the definition of the International Telecommunication Union (ITU-T Y.4000/Y.2060 (06/2012)): "A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving, interoperable information and communication technologies".

Nevertheless, as in any communication network, the IoT is exposed to various kinds of vulnerabilities and security threats. In particular, security is a critical challenge for the IoT development, as it constitutes an extended version of the conventional unsecured Internet model and combines multiple technologies such as Wireless Sensor Networks (WSNs), optics networks, mobile broadband, and 2G/3G communication networks. Each of the aforementioned technologies is prone to various security risks. Moreover, the objects in the IoT have the ability to interact with their environment automatically and autonomously, without any control of external factor and for this reason, various security and privacy issues can be caused. Finally, the multiple interconnections either between the users and objects or among objects generate tremendous amounts of data that are difficult to manage.

For the reasons above, many studies have examined the security issues in the IoT. Some of them determine the security requirements and challenges that the IoT generates [8–11]. Other studies identify the possible threats, vulnerabilities and countermeasures [12–14]. Furthermore, many papers examine the security issues of the IoT protocols [15–19], while others focus on specific security mechanisms and processes that can mitigate the possible cyberattacks [20–23]. Some of these cases are described briefly in the next section. Although these works provide significant and useful efforts, the ongoing evolution of the cyberattacks requires the simultaneous study of sufficient solutions, thereby making comprehensive survey papers necessary and valuable.

In this paper, we aim at providing a comprehensive analysis of the IoT. After determining

the security requirements and the challenges that the IoT generates, we focus our attention on the potential threats and the corresponding countermeasures. In particular, we introduce a risk assessment model, through which we evaluate qualitatively and quantitatively each possible threat per layer basis. Subsequently, utilizing this risk analysis, we examine and identify the appropriate countermeasures by paying special attention to the security mechanisms and vulnerabilities of the communication protocols. Finally, based on this analysis, we distinguish the research gaps in this field and provide directions for future work.

More specifically, the rest of this paper is organized as follows: Section 2 presents the motivation and contribution of this work. Section 3 provides an overview of the IoT technology. Section 4 and Section 5 determine the security requirements and challenges respectively. Section 6 analyzes and evaluates qualitatively and quantitatively the possible security threats per layer. Section 7 identifies and examines suitable countermeasures. Section 8 discusses the findings of our analysis and introduces directions for future work. Finally, Section 9 gives the concluding remarks of this paper.

2. Motivation and Contribution

Several research efforts have examined the security issues in the IoT, by analyzing the security challenges, threats and countermeasures. Some of them are listed in [8–11, 11–14, 24]. More concretely in [8–11, 24], the authors analyze the security requirements and identify the possible challenges. Accordingly in [11–14], the authors examine the potential security threats and the corresponding countermeasures. Other works follow a more precise approach, by investigating the security issues of the communication protocols [15–19]. In [15] J. Granjal et al. provide a comprehensive security analysis of several IoT protocols. In more detail, they examined the security issues of IEEE 802.15.4, IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN), IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), Datagram Transport Layer Security (DTLS) and Constrained Application Protocol (CoAP). In [16] K.T Nquyen et al. discuss the security attributes of various key protocols. In [17] the authors investigate the security attributes of Bluetooth Low Energy (BLE), ZigBee, LoRaWAN and Z-Wave. As in the previous case, in [18] D. Celebucki et al. focus their attention on the wireless protocols and identify possible exploits for ZigBee, Z-Wave and BLE. Furthermore, other works [20–23] focus on specific security applications, such as authentication, access control and Intrusion Detection and Prevention Systems (IDPS). For instance, in [25] L.Chen et al. studied methods for enhancing the privacy protection and robustness of location-based systems in the IoT. [20, 23] provide instances about the authentication and access control systems in the IoT. Regarding the IDPS systems, in [21] the authors introduce a comprehensive survey. Finally, M. Ammar et al. in [22] provide a detailed security analysis of 8 IoT frameworks.

Undoubtedly, the aforementioned papers constitute significant and useful efforts concerning the security of the IoT, by providing valuable information and taxonomies. Specifically, a great work took place regarding the analysis of the threats, countermeasures as well as security issues of the IoT protocols. However, the articles dealing with the various security threats do not take into account the risk level, thus making it impossible to properly de-

termine the appropriate countermeasures. Moreover, most of the papers that examine the security issues of the IoT protocols do not analyze their security features in detail, thus making it impossible to identify possible vulnerabilities, limitations and corresponding solutions. Only the [15] exhaustively investigates the security issues of various IoT protocols and determines the relevant security gaps. Nevertheless, this article is limited only to the IEEE 802.15.4 protocol without taking into consideration other wireless protocols at the Physical (PHY) and Medium Access Control (MAC) layers.

In this paper, we aim at covering the aforementioned deficiencies, by conducting a comprehensive security analysis of the IoT. In particular, at first, we determine the security requirements and identify the challenges that may affect the applicability and the implementation process of conventional security mechanisms. Then we define a risk assessment model which calculates qualitatively and quantitatively the risk level of potential threats. More specifically, this model rates each threat as (1) Low, (2) Medium and (3) High by taking into account the impact and probability of each invasion, as well as the existence of corresponding countermeasures. Each of the previous values is specified through precise limits. Subsequently, based on this model, we carry out an exhaustive analysis and evaluation of each possible threat in the IoT. Next, we investigate potential countermeasures, paying special attention to the security mechanisms of the IoT protocols. Concretely, we analyze in detail the security issues of IEEE 802.15.4, ZigBee, Z-Wave, BLE, LoRaWAN, RPL, Transport Layer Security (TLS), DTLS and CoAP. Finally, based on this analysis, we identify security gaps, vulnerabilities and limitations and provide directions for future work. The contribution of this paper is summarized in the following sentences:

- Comprehensive risk assessment and analysis of security threats in the IoT.
- Determination of appropriate countermeasures.
- Comprehensive analysis of the security issues of many IoT protocols (IEEE 802.15.4, ZigBee, Z-Wave, BLE, WAN, RPL, TLS, DTLS and CoAP).
- Determination of security gaps, vulnerabilities and limitations of the aforementioned IoT protocols.
- Providing directions for future research work.

3. IoT Overview

In this section, we aim at providing a brief overview of the IoT technology, by presenting its entities and the potential communication architectures. To analyze the security threats and identify the corresponding countermeasures, we adopt a four-layer architecture which consists of (1) perception layer, (2) communication layer, (3) support layer and (4) business layer.

3.1. IoT Devices

As mentioned before, the IoT can encompass many communication networks in which the devices can interact with each other via the Internet. These devices are usually called as "things" or "entities" and as illustrated in Fig. 1, they possess specific properties which are analyzed further below [26].

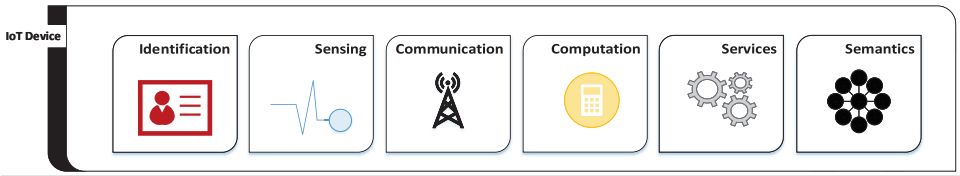

Figure 1: The properties of the IoT devices [26].

- **Identification**: Each IoT device needs to possess an identity, such as an Internet Protocol version 6 (IPv6) address in order to communicate with other objects [26].
- **Sensing**: The sensing methods are employed to obtain information from the physical environment [26].
- **Communication**: Communication refers to the interconnection methods which are utilized in order to communicate the objects with the users or with other objects [26].
- **Computation**: The computation methods are used to process the information which is obtained from the objects [26].
- **Services**: Services refer to the functions which are provided by the objects to the users in accordance with the information which they receive from the physical environment [26].
- **Semantics**: Semantics implies that the objects in the IoT have the ability to take the right information from an environment and provide this information as services at the appropriate time [26].

Examples of these devices, are Arduino [27, 28], Beagle Board [28, 29], Rasberry Pi [28, 30], CubieBoard [28, 31], and Radio Frequency Identification (RFID) tags [32]. These development boards include microcontrollers (MCUs), which contain a processor, a Read Only Memory (ROM), a Random Access Memory (RAM) and a number of both digital and analog general purpose input/output pins. Furthermore, these devices usually include various sensors, which are frequently hardwired in the MCU for local processing, responsive actuation, and relay to other systems. Examples of these sensors are temperature sensors, accelerometers, air quality sensors, potentiometers, proximity sensors, moisture sensors, and vibration sensors. Finally, they require a Real-Time Operating System (RTOS) for the information processing, memory management as well as utility services supporting messages and other communications. The selection of each RTOS is based on needed performance,

security and functional requirements of the product. Some popular IoT operating systems are TinyOS [33, 34], Contiki [34–36], Mantis [34, 37], FreeRTOS [34, 38], BrilloOS [34, 39] and ARM's mbedOS [34, 39].

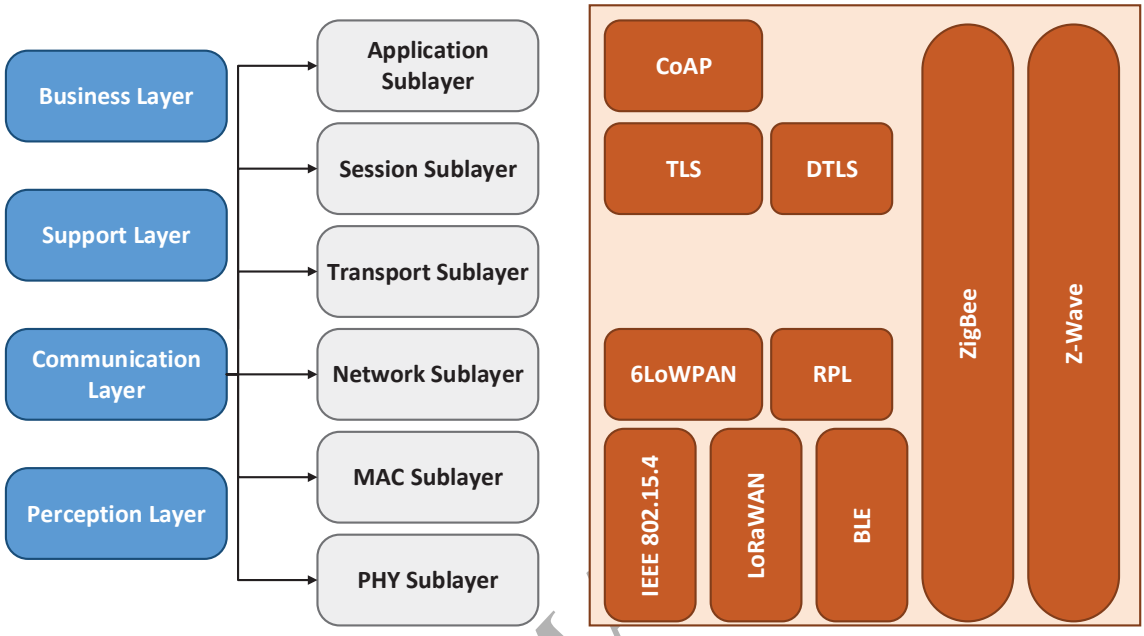

Figure 2: IoT Communication Architecture and Protocols.

3.2. Communication Architectures in the IoT

Like the traditional Information Technology (IT) networks, the IoT is divided into communication layers. However, a standard architecture has not been specified yet [26]. Many research efforts have suggested their own models, including three, four or five layers [26]. As illustrated in Fig. 2, we adopt and analyze the security issues of the IoT in a four-layer architecture which consists of: (1) Perception Layer, (2) Communication Layer, (3) Support Layer and (4) Business Layer. The perception layer includes the IoT devices that, as mentioned before, comprise technological elements such as sensors and actuators in order to sense the physical environment. The communication layer undertakes the reliable transmission of information among the other layers [8]. We consider that the communication layer consists of 7 sublayers: (1) PHY, (2) MAC, (3) transport (4) network, (5) transport, (6) session and (7) application. The support layer enhances the operation of the other layers, providing storage and computing services. The main technologies of this layer are the cloud and fog/edge computing. Finally, the business layer includes the software applications which are developed based on the user needs and the industry specifications.

4. Security Requirements in the IoT

Before evaluating the possible security threats in the IoT paradigm, firstly we should determine the corresponding security requirements. Many studies have investigated and determined the security requirements for IoT [13, 16, 40]. Based on them, we define the following security principles.

- **Confidentiality**: This term covers two related concepts. First, it signifies that unauthorized services must not access private information. Secondly, it assures the protection of privacy and proprietary information.
- **Integrity**: Integrity means that information and the IoT devices cannot be modified or utilized, by unauthorized users and objects.
- **Availability**: Availability implies that the computing resources and information should be available when they are needed by a service. This means that the IoT devices which are utilized to sense the physical environment, the computing systems that are used to store and process the information and the communication channels must operate properly.
- **Authenticity**: Authenticity assures that the information and transactions are genuine. In more detail, this principle must validate that the parties that participate in a transaction must be the ones whom they claim to be.

5. Security Challenges in the IoT

The security in IoT is characterized by high priority research interest since it is an evolution of the traditional, unsecured Internet model where the communications in the digital world meet the physical world. In particular, the security mechanisms in the IoT have to address the traditional networking attacks and at the same time, they have to offer secure communications for both type of interactions: human-to-machine and machine-to-machine. In order to fulfill the aforementioned security requirements and specify appropriate countermeasures, the following challenges have to be addressed.

- **Interoperability**: The development and the use of security mechanisms in the IoT should not largely limit the functional capabilities of the IoT devices.
- **Resource constraints**: The devices in the IoT are characterized by constrained resources in memory and computation; therefore, they may not support the expensive operations of the conventional security measures, such as the asymmetric encryption.
- **Resilience to physical attacks and natural disasters**: The IoT devices are typically small with limited or no physical protection. For instance, a mobile or a sensor device could be stolen, and the fixed devices could be moved or destroyed by natural disasters.

- **Autonomic control**: The traditional information systems require the users to configure them. However, the IoT devices have to establish their settings autonomously.
- **Information volume**: Many IoT applications such as the smart grid and smart city process a huge volume of sensitive and personal information, which is a potential target of an ever-increasing number of security threats.
- **Privacy protection**: Typically, the IoT devices include sensitive data which must be secured and not be identifiable, traceable and linkable.
- **Scalability**: The IoT networks usually involve an enormous number of objects. Therefore, the security and privacy protection mechanisms should be able to scale.

6. Security Threats in the IoT

The multiple interconnections and the heterogeneity of devices and technologies in the IoT generate possible cyber-physical security vulnerabilities that can be exploited by various cyberattackers. On the one side, sophisticated kinds of cyberattacks, such as zero-day attacks have been largely evolved having the capability to cause disastrous consequences in the human ecosystem. For instance, in December 2015 a Ukraine power grid was attacked, and electricity knocked out for 225.000 people [41]. Furthermore, on the other side, the hacking toolkits have been largely automated so even a novice can execute destructive cyberattacks. This section aims at analyzing and evaluating the potential cyber-physical attacks for each layer of the aforementioned IoT stack. The next subsections firstly introduce our risk assessment model and subsequently examine and evaluate each possible threat per layer.

Table 1: Probability of Threat.

No	Value	Rate
1	Rare	2.5
2	Unlikely	3.5
3	Moderate	4.5
4	Possible	6.5
5	Very Possible	7.5
6	Almost Certain	7.5

Table 2: Impact of Threat.

No	Value	Rate
1	Unimportant	2
2	Minor	3
3	Moderate	4
4	Significant	5
5	Destructive	6
6	Doomsday	7

Table 3: Rating of Countermeasures.

No	Value	Rate
1	There are available countermeasures	5
2	There are not available countermeasures	0

Table 4: Risk Level.

No	Value	Rate
1	Low	≤ 1
2	Medium	≥ 1 AND < 2
3	High	≥ 2

6.1. Risk Assessment Model

To examine and evaluate the possible threats, firstly we have to introduce and explain our risk assessment model. This process identifies the risk level of each threat based on the Equation 1. The *Probability* variable identifies the probability of the specific threat being achieved, while the *Impact* variable determines how destructive can be this threat. Finally, the *Countermeasures* variable denotes whether there are possible security solutions or not. The Tables 1-4 define the possible values for each of these variables.

$$Risk\ Level = \frac{(Probability \times Impact) - Countermeasures}{15} \quad (1)$$

6.2. Security Threats at the Perception Layer

For the IoT networks, the purpose of physical security is to protect the IoT devices that manage the information of the physical environment. In particular, the physical security includes two complementary requirements. Initially, it must prevent the damages in the physical infrastructure and secondly, it must prevent misuse of the physical infrastructure that can lead to the abuse or damage of the sensitive information. The main threats that dominate at the perception layer are listed below. Based on the previous risk assessment model, Table 5 and Fig. 3 provide a qualitative and quantitative evaluation of these threats respectively.

- **Natural disasters and environmental threats**: The natural disasters such as tornadoes, hurricanes, earthquakes, ice storms, lightning, and floods could destroy the physical infrastructures of the IoT networks. Also, environmental threats like inappropriate values of temperature and humidity, water accidents (e.g., electrical short circuit), fires, chemical accidents and infestations from living organisms (e.g., insects, rodents) could cause significant damages in the IoT networks. Consequently, this kind of threat results in the destruction of services, making impossible their availability. The impact of these threats can be characterized as "Doomsday"; nevertheless, their probability is "Rare", because such phenomena are very seldom and there are existent security mechanisms that can detect and mitigate them. Therefore the risk level of these threats is "Low".

- **Human-caused physical threats**: Human-caused physical threats are more challenging to address in comparison with the mentioned natural disasters and environmental threats, since they are specially designed to overcome protection measures and at the same time they target on the most vulnerable point of the physical infrastructure. Eavesdropping, vandalisms, device tampering, and misuse fall into this category. This kind of threats can affect all the aforementioned security requirements. Their impact can be considered as "Doomsday"; however they are "Unlikely" to happen because as in the previous case there are appropriate security measures. Hence, the level of this attacks is rated as "Medium".

Table 5: Qualitative Evaluation of Security Threats at the Perception Layer.

Security Threat	Probability	Impact	Countermeasures	Risk level
Natural disasters and environmental threats	Rare	Doomsday	✓	Low
Human-caused physical threats	Unlikely	Doomsday	✓	Medium

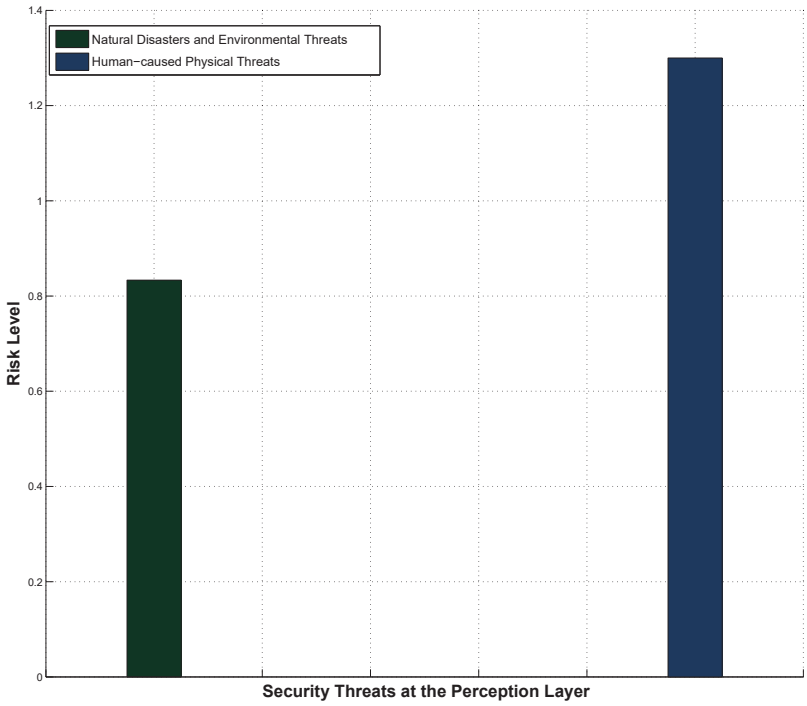

Figure 3: Quantitative Evaluation of Security Threats at the Perception Layer.

6.3. Security Threats at the Communication Layer

Both academia and industry had anticipated the security issues in the IoT and on this ground, it integrated appropriate authentication and encryption mechanisms in the IoT protocols. However, despite the existence of these mechanisms, the addressing of the network attacks is still an important security challenge [42]. Subsequently, we examine the most significant of these attacks. Moreover as in the previous case, based on the previous risk assessment model, Table 6 and Fig. 9 present a qualitative and quantitative analysis of these threats respectively.

- **Jamming attacks**: A jamming attack hinders the nodes to communicate with each other occupying the communication channel. More specifically, this kind of attack can be divided into four categories: constant jamming, deceptive jamming, random jamming, and reactive jamming [43, 44]. In general, the MAC protocol permits the

authorized nodes to transmit packets only if the corresponding communication channel is not used. However, in the case of the constant jamming attack, the attacker seeks to continuously utilize the communication channel by emitting a radio signal. Therefore the legitimate nodes cannot utilize the channel. On the other hand, in the deceptive jamming attack, the attacker continually sends packets to the communication channel without any pause. Hence, during the attack, a legitimate node is compelled to remain in receiving mode because it believes that there are remaining packets to receive. The random jamming model pursues to take into consideration the energy conservation. Specifically, the attacker possesses the ability to operate either in non-active state or in jamming state. The operation of the jamming state is based on one of the previous two models. In contrast with the previous models, the reactive jamming model utilizes an alternative strategy, in which it operates in a quiet mode when the communication channels are not used. However, when it realizes that there is a network activity in a specific communication channel, then it starts immediately to transmit a radio signal. A significant advantage of this model is that it is more challenging to detect. This kind of attacks targets the availability of the information and services. Their impact is "Significant" since they can destroy the IoT devices and be the first step for other attacks. They are considered as "Very Possible" because most of the existent countermeasures cannot fully prevent them. Consequently, the risk level is "High".

- **Selective forwarding attacks**: In selective forwarding attacks, the malicious nodes decline to transmit some packets, in order to destroy the routing paths of the network [42, 45]. Fig. 4 illustrates an instance of these attacks, in which node Z arbitrarily drops the packets coming from the nodes A and D. There are various kinds of these attacks. A typical case is the blackhole attack, in which the malicious node rejects each packet and does not forward any of them [46]. Another type is the Neglect and Greed attack, in which the attacker drops some packets or segments of them [47]. These attacks aim at destroying the availability of the information and services. Their impact and probability are "Significant" and "Possible" respectively. However, there are effective security mechanisms, such as IDPS systems that can detect and prevent this type of threats. Therefore, the risk level of these attacks is "Medium".

- **Sinkhole attacks**: In sinkhole attacks, the purpose of the malicious nodes is to direct the network traffic to a specific node. Usually, they advertise a particular route of the network and attracts the other nodes to utilize this route [42, 46, 48]. Fig. 5 presents a sinkhole attack, in which node E advertises itself. Nodes A, B, K, and Z are affected by the offensive and transmit the network traffic to node E. This kind of attack may not be able to cause significant damages to the network functionality, but it may be destructive when combined with other attacks [42]. As in the previous case, also the target of these attacks is the availability of systems. Their impact is "Significant", while their probability can be considered as "Possible". Nevertheless, IDPS systems are able to detect and prevent these attacks, hence the risk level is characterized as "Medium".

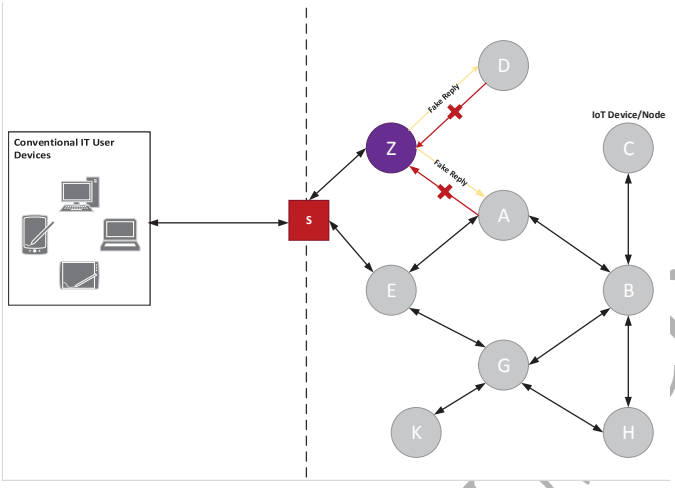

Figure 4: Selective Forwarding Attack.

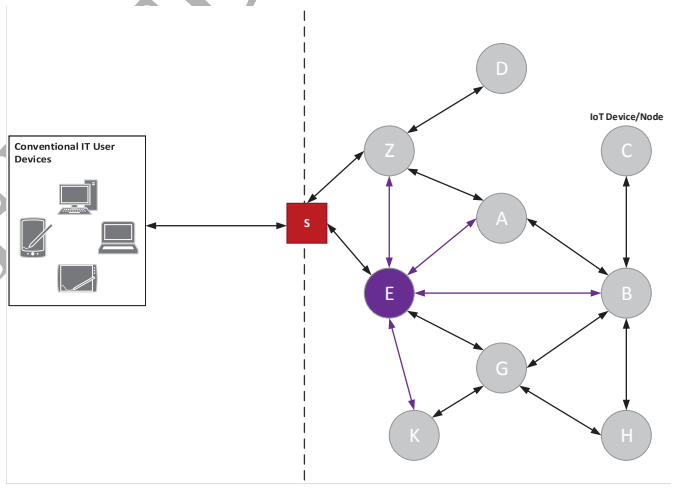

Figure 5: Sinkhole Attack.

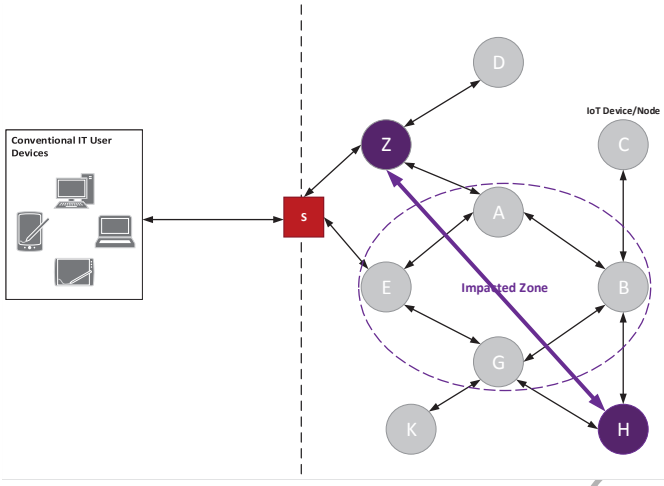

Figure 6: Wormhole Attack.

- **Wormhole attacks**: In wormhole attacks, the malicious nodes create a direct communication link which is utilized to forward the network traffic data ignoring the intermediate nodes [49]. This communication link is named wormhole and is characterized by exceptional network metrics such as high throughput. Usually, two collaborating nodes are needed to establish a wormhole. It is worth mentioning that such a connection could be utilized for specific significant reasons without malicious purposes [42]. However, when it is combined with other network attacks, such as a sinkhole attack or a sybil attack, then it constitutes a severe threat. Fig. 6 illustrates this state, in which nodes H and Z are connected through a wormhole link. The Wormhole attacks target the availability of services. Their impact is "Significant", while the probability to occur is "Possible". IDPS systems, as well as visualization mechanisms, can detect this kind of threats. Consequently, the risk level of these attacks is "Medium".

- **Sybil attacks**: In the sybil attack, the malicious nodes forge or create multiple identities in order to mislead other nodes [50, 51]. The purpose of the attacker, in this case, is to take under control different areas of the network, without using any physical node. In more detail, this attack can be classified into three types: SA-1, SA-2, and SA-3 [50]. A general model of the sybil attack is presented in Fig. 7, where nodes X, Y, and Z forge the identities of various nodes. Specifically, node K copies the identities of nodes A and D; similarly, node B utilizes the identities of nodes C and E; and finally, node Z forges the identities of nodes E and G. The target of these attacks is the authenticity and availability of the systems and services. Their impact can be considered as "Significant", while their probability to happen is "Possible". As in the previous cases, IDPS systems can address these attacks. Thus, the risk level is "Medium".

- **HELLO flood attacks**: Typically, a node utilizes HELLO messages to join a network; however, a malicious node can employ the specific messages in order to mislead other nodes aiming to identify it as a neighbor. In RPL networks, the DODAG Infor-

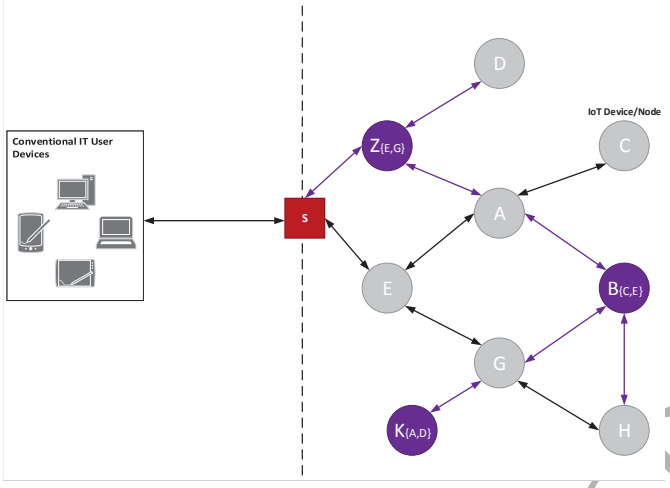

Figure 7: Sybil Attack.

mation Object (DIO) messages can be utilized with strong routing metrics in order to start such an attack [42]. Fig. 8 represents the first stage of a HELLO flood attack, where node D transmits HELLO messages to the other nodes in the network. It is worth mentioning, that this attack can endure for a limited time in the RPL networks, since the RPL protocol includes defense mechanisms that are able to hinder this kind of attacks. These attack target mainly on the authenticity of the systems and secondly the availability of the information and services. Their impact and probability is "Moderate" and "Possible" respectively. However, the communication protocols, such as RPL can address them efficiently. Therefore, their risk level is "Medium".

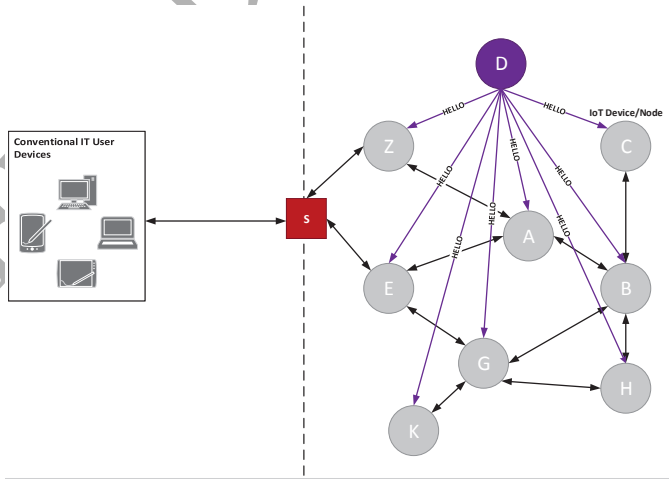

Figure 8: HELLO Flood Attack.

- **Traffic analysis attacks**: Traffic analysis is the procedure of capturing and analyzing network packets in order to gather significant data, such as network flows or the payload of decrypted packets. There are many traffic analysis software packages, such

as Wireshark [52–54], Tcpdump [52, 55], Kismet [56] and Scapy [57]. More specifically, this kind of software is divided into two components: a sniffer and a protocol analyzer. The sniffer captures a copy of the transmitted network packets, while the protocol analyzer undertakes to decode and analyze these packets. The target of these attacks is the confidentiality of information. They are "Very Possibly" to happen, while their impact can be considered as "Moderate". Encryption mechanisms can prevent the consequences of such kind of attacks. Consequently, their risk level is "Medium".

- **Man-in-the-middle (MiTM) attacks**: This kind of attack is defined as a form of eavesdropping in which the intruder can illegally monitor the communication messages that are exchanged between two parties. Examples of these attacks are Neighbor Discovery Protocol (ND or NDP) poisoning [58, 59], Address Resolution Protocol (ARP) poisoning [60, 61], replay attacks [62, 63], session hijacking [64, 65] and malicious proxy servers [66, 67]. These attacks threaten at the same time the confidentiality and authenticity of the systems. They are "Very Possibly" to occur while their impact is "Significant". Encryption mechanisms and IDPS systems can prevent these attacks. Their risk level is "High".

Table 6: Qualitative Evaluation of Security Threats at the Communication Layer.

Security Threat	Probability	Impact	Countermeasures	Risk level
Jamming Attacks	Very Possible	Significant	✓	High
Selective Forwarding attacks	Possible	Significant	✓	Medium
Sinkhole Attacks	Possible	Significant	✓	Medium
Wormhole Attacks	Possible	Significant	✓	Medium
Sybil Attacks	Possible	Significant	✓	Medium
Hello Flood Attacks	Possible	Moderate	✓	Medium
Traffic Analysis Attacks	Very Possible	Moderate	✓	Medium
MiTM Attacks	Very Possible	Significant	✓	High

6.4. Security Threats at the Support Layer

As mentioned above, the key technologies of the support layer are the cloud [68, 69] and fog computing [70, 71]. However, the use of them raises a number of security issues, particularly in the area of database security. The main security threats in this layer are listed below. Furthermore, Table 7 evaluates these threats qualitatively, while Fig. 10 quantitatively.

- **Unauthorized access**: The unauthorized access refers to stealing the credentials of legitimate accounts or illegally utilizing the resources of an organization. Therefore, the unauthorized users have the ability to access significant information and compromise the security requirements that we discussed before. Usually, the target of these attacks is the confidentiality and authenticity. Their probability is "Possible" while

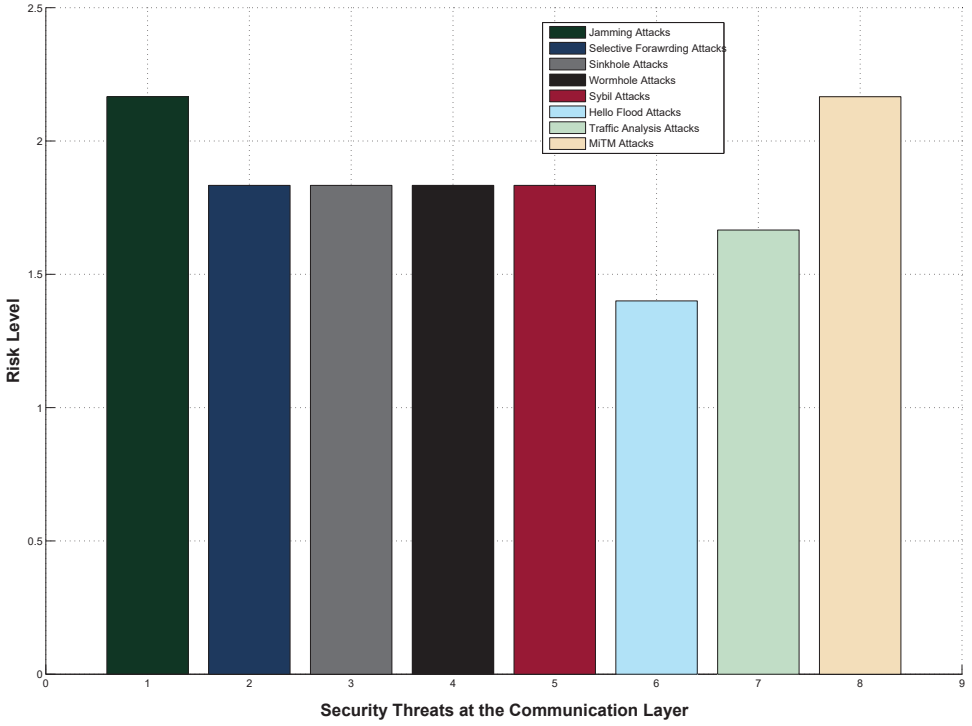

Figure 9: Quantitative Evaluation of Security Threats at the Communication Layer.

their impact is considered as "Significant". Effective authentication and access control mechanisms can prevent these attacks. Hence, their risk level is "Medium".

- **Malicious insiders**: From the nature of the cloud and fog/edge computing, the users are obliged to grant an unusual level of trust onto the cloud provider. Hence, a significant risk is the actions of the malicious insiders [72, 73]. This attack can affect all the security requirements. Their probability is "Possible" while their impact is considered as "Destructive". The potential security solutions maybe cannot prevent these attacks. Hence the risk level is "High".

- **Insecure software services**: The cloud computing technology provides services such as, web applications, operating systems, Application Programming Interface (API) and virtual machines creation. However, there is a likelihood that these services may be compromised by malware. In general, the security requirements of a cloud computing technology depend on the protection of the provided services [74]. This kind of attack can affect all the security requirements. Their impact can be "Significant" however, they are "Unlikely" to occur. Through the secure programming, these attacks can be avoided. Consequently, their risk level is "Low".

- **Unknown risk profile** [74]: This layer includes services that may be provided and controlled by external entities. Therefore, these entities are responsible for multiple

functionality issues that comprise, among others the security of applications and information. For instance, an organization or a company could use the storage services of an independent cloud computing provider. Hence, the security and privacy of the information which is stored in the specific services are controlled by an external provider. This threat is not specific. Consequently, it can simultaneously affect all the security requirements. However, its probability is "Rare" and the corresponding impact can be considered as "Moderate". Based on this information, the risk level, in this case, is "Low".

Table 7: Qualitative Evaluation of Security Threats at the Support Layer.

Security Threat	**Probability**	**Impact**	**Countermeasures**	**Risk level**
Unauthorized Access	Possible	Significant	✓	Medium
Malicious Insiders	Possible	Destructive	✓	High
Insecure Software Services	Unlikely	Significant	✓	Low
Unknown Risk profile	Rare	Moderate	X	Low

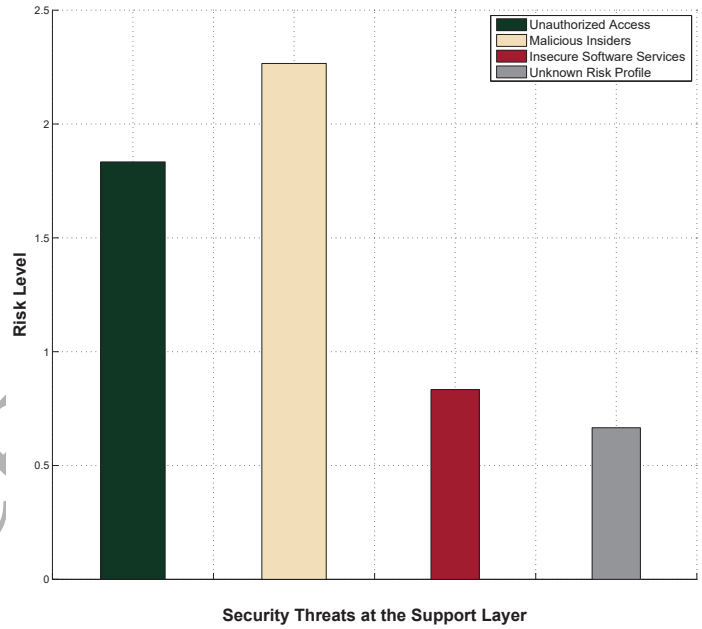

Figure 10: Quantitative Evaluation of Security Threats at the Support Layer.

6.5. Security Threats at the Business Layer

As mentioned before, this layer provides services according to the user's needs. Therefore, the main threats in this layer include social engineering techniques [75, 76] and the exploit of

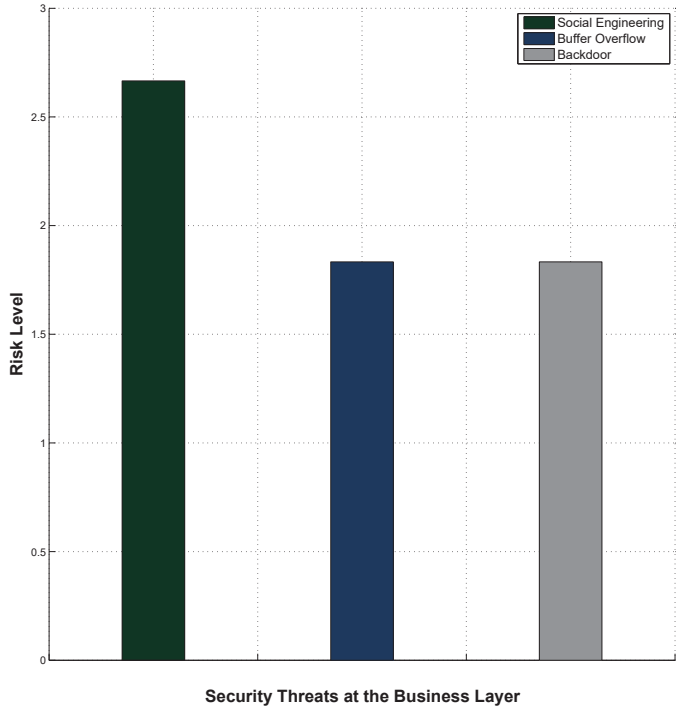

Figure 11: Quantitative Evaluation of Security Threats at the Business Layer.

programmatic gaps in these services. Subsequently, we analyze further these threats. As in the previous cases, Table 8 identifies the risk level of these threats in a qualitative manner, while Fig. 11 quantitatively.

- **Social engineering**: Social engineering is a psychological attack which aims to mislead users in order to disclose confidential information or unwittingly perform specific malicious activities [75, 76]. The most prolific social engineering technique is the phishing attack in which the intruder pursues to gain the trust of the user by using spoofed emails, instant messages or Domain Name System (DNS) spoofing processes. Usually, the users are directed to a fake website which urges them to insert sensitive information. A more hazardous variation of this kind of attack is the spear phishing. In this case, the attacker has examined the recipients thoroughly, and each fake message is carefully crafted in order to suit with recipient profile [74]. These attacks mainly focus on the confidentiality, integrity and authenticity of information. They are "Very Possibly" to occur while their impact can be "Destructive". Security management and education processes are possible countermeasures against these attacks. Based on the previous information, their risk level is considered as "High".

- **Buffer overflow**: According to NIST, a buffer overflow or differently buffer overflow or buffer overrun is a kind of attack which permits the intruder to insert more data in a buffer than the capacity limit allows. The attacker aims to overwrite the existing information in the buffer in order to insert malicious code that will make it possi-

ble to control the overall system [77, 78]. Some examples of these attacks are stack overflow, global data area overflow, format strings overflow, heap overflow and integer overflow. Usually, the attacker utilizes assembly code to execute such an attack. These attacks aim at compromising the integrity and authenticity of systems. Their impact is "Significant", while the probability to occur is "Possible". An efficient countermeasure against this threat is the secure programming. Their risk level is calculated at "Medium".

- **Backdoor**: A backdoor or differently trapdoor is a code segment in software that enables an intruder to overcome specific processes that can include security controls [74]. More precisely, a backdoor is activated when the user employs particular credentials, or a specific sequence of events is performed. It is noteworthy that a backdoor is not necessarily a security threat, as a system administrator can utilize them to overcome time-consuming procedures and control the functionality of the software expeditiously. However, extremely adverse effects can be caused if an attacker is aware of the specific block of code. The malicious backdoors usually operate as a network service which enables the attacker to connect to an unusual network port and executes malicious activities. The target of backdoors usually is the confidentiality, integrity and authenticity of information. Their impact is "Significant", while the probability to happen is "Possible". As in the previous case, the safe programming is the solution for this threat. Consequently, their risk level is calculated at "Medium".

Table 8: Qualitative Evaluation of Security Threats at the Business Layer.

Security Threat	**Probability**	**Impact**	**Countermeasures**	**Risk level**
Social Engineering	Very Possible	Destructive	✓	High
Buffer Overflow	Possible	Significant	✓	Medium
Backdoor	Possible	Significant	✓	Medium

6.6. Multi-Layer Security Threats

In this subsection, we consider cyberattacks and malware that can be performed in multiple layers. More specifically, we distinguish the following cases. Moreover, Table 9 and Fig. 12 calculates their risk level qualitatively and quantitatively.

- **Cryptanalytic attacks**: Cryptanalysis [79, 80] can be defined as the process in which the attacker attempts to discover the original message (plaintext) from the scrambled message (ciphertext). These attacks can be executed in all layers and target the confidentiality, integrity and authenticity of information. Some kinds of these attacks are Ciphertext-only attack, Known-plaintext attack, Chosen-plaintext attack, Chosen-ciphertext attack and Side-channel attack. Their impact can be "Destructive"; however, they are "Unlikely" to happen. The encryption mechanisms of the communication protocols can address these attacks. Consequently, their risk level is "Medium".

- **Denial-of-service (DoS) attacks**: The DoS attacks [81] target the availability of the computing systems and they can be performed in all layers of the proposed IoT stack. In particular, they pursue to hinder the legitimate entities to utilize applications or services by weakening the computing resources that support them. For instance, they attempt to reduce the performance of the Central Processing Unit (CPU) or to flood the memory size [82]. Some examples of this type of attacks are flooding attacks, distributed DoS attacks (DDoS), reflection attacks, amplification attacks and jamming attacks that were discussed before. These attacks are "Very Possible" to happen, while their impact can be "Destructive". IDPS systems can mitigate these attacks, but cannot fully prevent them. Hence, their risk level is "High".

- **Spyware**: Spyware is a type of malware that collects sensitive information such as, among others, keystrokes, screenshots, authentication credentials, network traffic, and internet usage habits from a system and transmits it to another system. It can be performed at the support and application layer and targets the confidentiality of the IoT system. Spyware is a type of malware that collects sensitive information such as keystrokes, screenshots, authentication credentials, network traffic, and internet usage habits from a system and transmits it to another system. It can be performed in all layers and target the confidentiality of information. Their impact is "Destructive", but they are "Unlikely" to happen, as encryption mechanisms and IDPS systems can hinder this malware. Therefore, their risk level is "Medium".

- **Botnets**: A bot (robot), or zombie or drone is a malware which aims to put under control the services of a computing system in order to utilize them for malicious activities. For instance, these services can be used to perform a DoS attack or to infect more systems. Ordinarily, the infected systems form groups that are named botnets and operate in a coordinated manner. The payload attacks of this malware can be performed in all layers and usually target the availability and authenticity of systems [83, 84]. Their impact can be "Destructive", while their probability to occur is "Moderate". The IDPS systems have the ability to address these attacks, by adopting anomaly detection processes. The risk level of this malware is "Medium".

- **Rootkit**: A rootkit [85] constitutes a set of services that are installed on a computing system and aim to provide uninterrupted and covert access with administrator privileges. Specifically, the administrator privileges enable the attacker to perform any action, such as modifying files or installing a backdoor. This type of malware can be performed in all layers and classified based on the following features: persistent, user mode, memory based, kernel mode and virtual machine based [74, 85]. They mainly aim at compromising the integrity and authenticity of systems. Their impact can be "Destructive", while the probability to happen is "Possible". IDPS systems and maintenance processes can detect and prevent these attacks. Their risk level is "High".

- **APTs**: Advanced Persistent Threats (APTs) [86–88] do not refer to a new malicious

software, but they are organized, persistent cyberattacks against specific targets that usually come from the political and business environment. They can be performed in all layers of the aforementioned IoT stack and commonly seek to steal significant information or to destroy the overall operation of the target including the physical infrastructures. Mainly, they are characterized by abundant computational resources, the precise target choice and the extended period of their implementation. Examples of this type of attacks are Stuxnet, Duqu, Flame, and Red October [89]. These attacks threaten all the security requirements. Their impact can be considered as "Doomsday", while their probability can be characterized as "Moderate". There are not sufficient countermeasures that fully prevent these attacks. Therefore their risk level is "High".

Table 9: Evaluation of Multi-Layer Security Threats.

Security Threat	**Probability**	**Impact**	**Countermeasures**	**Risk level**
Cryptanalytic Attacks	Unlikely	Destructive	✓	Medium
DoS Attacks	Very Possible	Destructive	✓	High
Spyware	Unlikely	Destructive	✓	Medium
Botnets	Moderate	Destructive	✓	Medium
Rootkit	Possible	Destructive	✓	High
APTs	Moderate	Doomsday	X	High

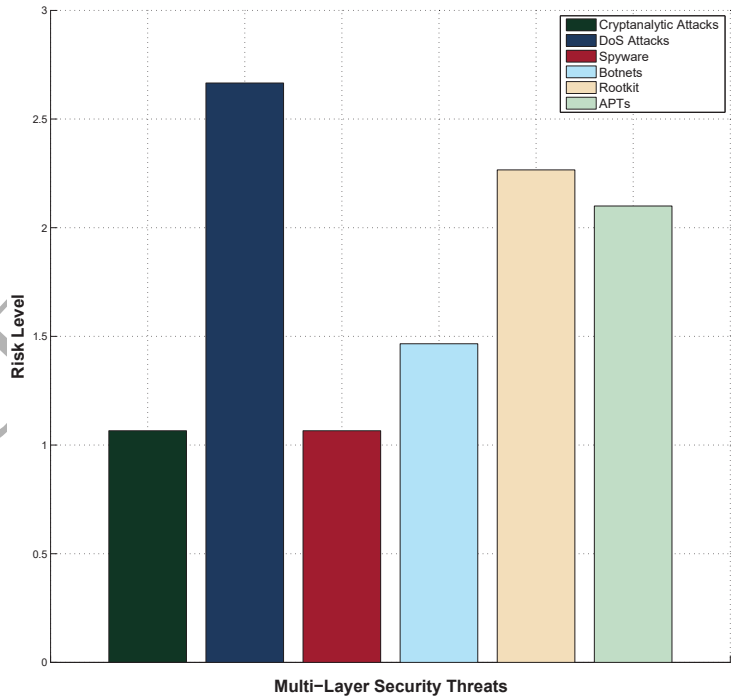

Figure 12: Quantitative Evaluation of Multi-Layer Threats.

7. Countermeasures

In this section, we investigate possible security solutions for the aforementioned threats. The ideal solution is the prevention of the possible threats; however, the specific goal is nearly impracticable, but appropriate countermeasures can mitigate the impact of these threats.

7.1. Countermeasures at the Perception Layer

The security mechanisms at the perception layer have to address the natural disasters, the environmental threats, the human-caused physical threats and the jamming attacks. Open Web Application Security Project (OWASP) has pointed out that the inadequate physical security remains in the top 10 of IoT vulnerabilities [13]. More specifically, on the one hand, specific technical approaches such as infrastructure design, sensor design and placement, mitigation procedures, personal training, and recovery mechanisms can efficiently handle the natural disasters and the environmental threats. On the other hand, in order to address the human-caused physical threats, the first step is to assure that only legitimate users and objects can access the physical devices and their information. Therefore, user authentication systems, physical access control mechanisms, and a trust framework are required. In more detail, user authentication mechanisms such as password-based schemes, token-based schemes (e.g., electronic keycards, smart cards) and static or dynamic biometric systems (e.g., recognition by fingerprints, retina, iris, facial characteristics, hand geometry, voice) determine if a user or an object can access the physical resources and their data. Access control mechanisms determine the access privileges of the authenticated users and objects. Finally, a trust framework should accompany the previous requirements in order to apply the need-to-know principle: "What do you need to know about someone in order to deal with them?".

7.2. Countermeasures at the Communication Layer

As mentioned previously, the IoT protocols [90] integrate important security mechanisms that guarantee the confidentiality, integrity and authenticity of the communications. In this subsection, we analyze the security mechanisms that are provided by the most used IoT protocols of the communication layer, and also we provide their limitations as well as corresponding solutions from the literature. More specifically, we discuss the security mechanisms that are implemented by IEEE 802.15.4, ZigBee, Z-Wave, BLE, LoRaWAN, 6LoWPAN, RPL, TLS, DTLS and CoAP. Table 10 provides on an abstract level the security capabilities and vulnerabilities of these protocols. Although most of the previous protocols satisfy the essential security requirements, the next subsections provides their limitations and possible vulnerabilities. Moreover, a subsection refers to the countermeasures against the jamming attacks, as this kind of attacks target on the availability of services and constitutes a common threat for many communication protocols. Finally, the firewall and IDPS systems are analyzed, since they have the ability to monitor and control all the communication sublayers.

7.2.1. IEEE 802.15.4 Security

The IEEE 802.15.4 protocol constitutes a common option for short-range communications in an IoT environment. Specifically, it controls the transmission of information at the PHY and MAC sublayers. At the PHY sublayer, it supervises the radio frequency, the energy consumption, the signal management and the determination of the communication channel. On the other hand, at the MAC sublayer, besides the data processing, it supports additional services, such as security mechanisms, node association and packets validation.

Although IEEE 802.15.4 is responsible for the communication management at the PHY and MAC sublayers, it includes security mechanisms only at the MAC sublayer. However, these security mechanisms are significant for the overall security of the higher protocols that conclude the architecture of the IoT communication stack. More specifically, the activation of the security mechanisms in IEEE.802.15.4 is a non-obligatory setting. The Frame Control field involves a bit called Security Enabled Bit (SEB) which determines the implementation of the security services included in the Authentication Security Header (ASH) field. ASH determines the combination of the security algorithms and also defines the key construction procedure for the symmetric encryption. The applications which require only information confidentiality at the MAC sublayer could utilize Advanced Encryption Standard (AES) in Counter (AES-CTR) security mode. On the other hand, the applications which demand both integrity and authenticity of data could employ AES in the Cypher Block Chaining (AES-CBC) security mode. Finally, the applications, which require confidentiality, integrity, and authenticity can utilize the combined Counter with CBC (AES-CCM) security mode. Finally, it should be noted, that the specific protocol comprises solutions against replay attacks and also supports access control capabilities. More precisely, the sender has the ability to break the original packet into 16-blocks, which are encrypted utilizing either a nonce or an Initialization Vector (IV). Concerning the access control mechanisms, IEEE 802.15.4 utilizes Access Control Lists (ACL) that can involve 255 records. These records determine the access privileges for security-related information, such as IEEE 802.15.4 addresses, security suite, encryption key, last IV and replay counter.

Despite the fact that the IEEE 802.15.4 protocol comprises important security services, it is characterized by some limitations. More specifically, it cannot protect acknowledgment messages (ACK) concerning integrity and confidentiality [15]. This constraint may lead the adversaries to forge acknowledgment messages and execute various kinds of DoS attacks. Also, the supported ACLs do not effectively manage the records which employ the same encryption key. This state could lead the sender to reuse the nonce value with the result when stream ciphers are utilized then there is a likelihood of recovering a plaintext from a ciphertext [15]. The RFC 5246 document [91] defines the term stream ciphers as follows: "In stream cipher encryption, the plaintext is exclusive-ORed with an identical amount of output generated from a cryptographically secure keyed pseudorandom number generator." In addition, in some cases, the re-employment of the nonce values is probable, if the ACLs data are erased. Finally, it is worth mentioning that the particular protocol cannot sufficiently implement all keying models [15]. Therefore, in conclusion, the aforementioned limitations offer future work possibilities.

7.2.2. ZigBee

The ZigBee technology was designed to provide an efficient and secure short-range communication in an IoT environment, paying attention to the energy consumption issue. More specifically, ZigBee introduces a communication stack which consists of: (1) PHY layer, (2) MAC layer, (3) network layer and (4) application layer [92]. In a ZigBee environment, there are three kinds of devices: coordinator, routers and end nodes. The coordinator device constitutes the core of a ZigBee network as it undertakes to establish and initialize the network, by identifying the communication channel, managing and configuring the privileges of the other entities as well as determining the security mechanisms. It should be noted that the coordinator device should be alive continuously and it determines whether a new device can join the network or not. On the other side, routers are responsible for the intermediate communication either between the coordinator and the end devices or among end devices. As in the previous case, the routers have to operate continuously as long as the ZigBee network is active. Finally, the end nodes are power constrained IoT devices whose role is to sense the physical environment. The end nodes can communicate with other devices only via a parent device (e.g. router) and also they have the ability to operate in sleep mode, thereby reducing the energy consumption.

Concerning the security issues, ZigBee utilizes the IEEE 802.15.4 protocol at the PHY and MAC layers and it itself provides the communication and security processes at the network and application layers. Therefore, the security of the MAC layer is based on IEEE 802.15.4 and specifically, it utilizes the AES-CCM* mode which is a modified version of AES-CCM [92]. More detailed, AES-CCM* offers the capability to select either encryption or authentication processes while both of them are applied in the AES-CCM mode [93]. ZigBee provides the capability to choose various security models [94]. The centralized security model offers the most sufficient security processes. Its functionality is based on the coordinator device which is called Trust Center (TC). In particular, in this model, five security keys are defined that are coordinated by the corresponding communication layer. The network key is a 128-bit key which is shared among all devices. ZigBee provides two security levels: (1) high-security and (2) standard security. In the first case, the network key is always encrypted when it is distributed among the devices. On the other hand, the standard security level does not introduce encryption processes which is a well-known vulnerability [93]. In the high-security level, a global link key undertakes to encrypt the network key, when it has to be transmitted from TC to the existing devices of the network. This key is predefined between the TC and the other devices and also is used during the joining process of a new device [94]. Similarly, a unique link key is utilized when the network key has to be sent from TC to a new a device which has not already joined the network. As in the previous case, also this key is predetermined between the TC and the new device [94]. Furthermore, the TC link key is responsible for the communication between TC and the other devices. This key is randomly generated by TC and replaces the previous key [94]. Finally, the application key is encrypted via the network key and is employed for the interaction between the routers and end nodes. This key is produced by TC [94]. All the aforementioned security keys can be transmitted over the air or be pre-established in the corresponding devices [93]. In addition it is noteworthy, that a Message Integrity Code (MIC) which is generated inside

the AES-CCM* mode guarantee the integrity and freshness of data. Finally, regarding the replay attacks, ZigBee devices can employ a frame counter which is increased whenever a frame is received, thereby rejecting the frames that do not meet with the specific counter. These counters are set to zero whenever the network key is updated [93].

Although, we described previously a security mechanism of ZigBee against replay attacks, it is possible to perform such type of attacks successfully by combining specific hardware and software equipment. More specifically, the AVR RZ Raven USB can be used either as a ZigBee Personal Area Network (ZPAN) or end node to sniff and capture the network traffic and accordingly the network key whether it is not encrypted [93]. Furthermore, Killerbee [95] is a software tool developed using the Python programming language and it can be used to capture and analyze the ZigBee traffic. In particular, it comprises the following modules (1) zbdsniff which can capture the network traffic and the network key whether it is not encrypted, (2) zbstumbler which constitutes a ZigBee network discovery tool and (3) zbassocflood which can flood a device with multiple connections [96]. In [93] the authors describe a sniffing attack, utilizing the aforementioned technologies. Another common attacks against ZigBee are DoS and specifically the jamming attacks that target on the battery lifetime of the end nodes. In [93] the authors describe theoretically a jamming attack against ZigBee called ZigBee End-Device Sabotage Attack. More detailed, they consider an attacker which represents a router or the TC, transmitting continuously request messages to the end nodes. This situation leads end devices to send reply messages thus exhausting the battery lifetime. Moreover, as in the previous cases, the unauthorized physical access can lead to critical states. Based on [93], in such a case, the attacker will be able to extract the security keys. Therefore, the research efforts should pay particular attention to the specific kind of attacks, deploying appropriate countermeasures. First of all, the standard security level should not be used, as it enables replay attacks. Subsequently, intrusion detection and prevention systems can be considered as an effective solution which will detect, mitigate or prevent these attacks timely. Finally, anti-jamming countermeasures and physical security mechanisms are necessary for the normal operation of such a network.

7.2.3. Z-Wave Security

Z-Wave is a proprietary technology, which is designed for short-range IoT communications. Z-Wave is provided by the Z-Wave Alliance, which includes more than 600 companies. Characteristic examples are Huawei and Siemens. In particular, this technology is deployed in a mesh network utilizing a four-layer architecture which on the basis of the OSI model includes: (1) PHY layer, (2) data link layer, (2) network layer and (4) transport layer [97]. It should be noted that PHY and the data link layer have been standardized as the G.9959 standard by ITU. Z-Wave is capable of connecting 232 devices. More specifically, the devices in a Z-Wave network are divided into two categories: (1) controllers and (2) slaves. The controllers are responsible for the network management by determining the corresponding specification and controlling whether a new device can join the network or not. Moreover, a primary controller defines a specific and unique home ID to the network. On the other hand, the slaves constitute typical IoT devices.

Based mainly on a network key, Z-Wave provides multiple security mechanisms, thus

enabling safe communications. More specifically, Z-Wave organizes the security mechanisms into two main classes: (1) Security 0 (S0) and (2) Security 2 (S2) [17]. Furthermore, S2 consists of three subclasses: (1) S2-Access Control subclass, (2) S2-Authenticated subclass and (3) S2-Unauthenticated subclass [97]. S2-Access control is considered the most secure option, while S2-Unauthenticated and S0 focus on very constrained and legacy devices respectively. The security of the previous classes and subclasses include AES-128 CCM mode encryption and authentication processes, except the S2-Unauthenticated subclass. Regarding the key exchange process, S2 enables the sharing only among the devices of the same subclass. For example, a device which belongs to S2-Access Control cannot exchange the network key with a device of the S2-Unauthenticated subclass. The key exchange process of the S2 class is conducted via the Curve25519 model which is considered a safe option. However, a side-channel attack against this model was recently discovered [98]. On the other side, the ECDH scheme is used for the key exchange process of the S0 class. Finally, S2 provides AES-128 Cipher-based Message Authentication Code (AES-128-CMAC) and predetermined nonces, thereby ensuring the integrity of data and the protection against the replay attacks respectively.

Z-Wave assures the confidentiality and integrity of information. So far, there have not been identified specific security issues against Z-Wave. There are only some successful exploits against specific implementations. In particular, Z-Wave allows the communication with legacy devices that may not include sufficient security mechanisms. This fact can lead to various security threats such as replay attacks. Moreover, although Z-Wave integrates AES-128 encryption processes, in many cases the manufacturers do not activate these mechanisms. In [99], the authors tested various Z-Wave devices and they argue that only 9 of 33 incorporates the available security measures. In [99], the authors demonstrate a successful cyberattack against a Z-Wave-based door lock application by exploiting a vulnerability of the key sharing process. This vulnerability is rated 6.5 in the Common Vulnerability Scoring System (CVSS) [18]. Finally, in [100] M. Smith provides a tool called EZ-Wave, which is capable of performing various penetration testing processes against Z-Wave. The efficiency of this tool is demonstrated by turning on and off various bulbs of a Z-Wave network thus leading to their destruction. This vulnerability is rated 6.5 in CVSS [18]. In conclusion, Z-Wave provides valuable security mechanisms that can largely guarantee the safe operation of the network. The manufacturers and vendors should always follow the security updates, configuring appropriately the corresponding devices. The IDPS systems can be a useful countermeasure against the potential cyberattacks. Moreover, the research efforts should focus on self-healing mechanisms, thus providing additional preventive measures for the protection of the critical facilities.

7.2.4. Bluetooth Low Energy Security

The BLE technology is a modification of Bluetooth to support short-range communications, especially for constrained IoT devices, providing them the ability to form wireless networks, called piconets [101]. Bluetooth was introduced with the formation of a non-profit consortium of many organizations and companies, called Bluetooth Special Interest Group (SIG). More specifically, BLE was generated from the Bluetooth 4.0 specification

and subsequently, the specifications 4.1 and 4.2 updated its features. The architecture of a BLE piconet mainly consists of two kinds of devices: (1) master nodes and (2) slave nodes. The master node is responsible for initiating the network, while the slave nodes are power-constrained devices sensing the physical environment. It is noteworthy, that a slave node can be a master node in a different piconet. A chain of piconets is named scatternet [101]. Moreover, BLE enables the existence of broadcasters and observers that periodically broadcast and listen messages respectively. Finally, BLE enables the communication up to 50m, while the maximum data rate is calculated to 1Mbps.

Based on [101], the security features of BLE focus on the authentication, confidentiality, integrity and pairing properties. The pairing term refers to the generation and storage processes of the secret keys that are used for the encryption and authentication mechanisms provided by BLE. There are three keys that should be distributed: (1) Long-Term Key (LTK), (2) Identity Resolving Key (IRK) and (3) Connection Signature Resolving Key (CSRK). LTK is used for the encryption mechanisms. IRK and CSRK are responsible for determining private addresses and data signing respectively. It should be noted that LTK is divided into Master LTK (MLTK) and Slave LTK (SLTK). In particular, two security modes are defined. The first security mode (Security Mode 1) includes four levels. The first level does not integrate any security mechanism. The second level encompasses encrypted communication, but it does not require authenticated pairing. The third level requires both authenticated pairing and encryption processes. Finally, level 4 introduces upgraded encryption and authentication processes, called Secure Connections. On the other side, the second security mode (Security Mode 2) comprises two levels that are related with the signing processes. Specifically, the first level defines data signing with non-authenticated pairing, while the second demands authenticated pairing and data signing. Regarding the pairing process, there are four models: (1) Numeric Comparison, (2) Passkey Entry, (3) Just Works and (4) Out of Band (OOB). The devices that follow the specification 4.0 and 4.1 use a legacy pairing process in which the devices firstly, utilize a temporal key (TK) to exchange some random values and then based on TK and these random values, they generate a Short Term Key (STK) which is used to distribute securely LTK, IRK and CSRK. On the other side, the devices that follow the specification 4.2 use a Secure Connections pairing process, in which the LTK is not distributed but is generated autonomously in each device utilizing AES-128-CMAC. Subsequently, this LTK is used to distribute securely IRK and CSRK. It is noteworthy, that in contrast to specifications 40 and 4.1, the specification 4.2 enhances the security of the pairing process through the addition of AES-128-CMAC as well as P-256 Elliptic curve. Finally, concerning the confidentiality of data, BLE utilizes AES-CCM, while there is not an explicit authentication mechanism, as the encryption of the link satisfies the authentication process.

BLE presents various security vulnerabilities. In particular, the privacy of users and BLE devices can be compromised, if an attacker is able to associate the address of a device with a specific user [101]. Moreover, replay attacks constitute a possible threat, since the attackers can violate the legacy pairing process, by capturing LTK, IRK and CSRK. In [102], the authors demonstrate this vulnerability, by predicting and identifying TK within 20 seconds. This vulnerability is rated with 7.4 in CVSS [18]. Furthermore, a crucial issue

is that the first level of Security Mode 1 does not incorporate any security mechanism [101]. In addition, although the specification 4.2 introduced efficient processes that ensures many security requirements, the manufacturers and vendors have the ability to select the level of security thus can lead to various security accidents [17]. Certain vendors have attempted to deploy encryption mechanisms at the application layer; nevertheless, the MiTM attacks are a severe security issue [103, 104]. Finally, the specifications themselves are characterized by high complexity, thereby resulting in several security issues and vulnerabilities [17]. Consequently, on the basis of the above analysis, we consider that vendors and manufacturers that follow the specification 4.2 should employ Security Mode 1 Level 4, while they that follow the specifications 4.0 and 4.1 should employ the Security Mode 1 Level 3.

7.2.5. LoRaWan Security

The LoRaWan technology was adopted in an effort to enhance the functionality of Low Power Wide-Area Networks (LPWAN) regarding mainly the consumption capability, storage capacity, long-range communication and transmission cost. Its architecture is based on four main entities: (1) end nodes, (2) gateways, (3) network server and (4) application server. The end nodes are usually IoT devices that collect information from the physical environment and transmit them to gateways via the LoRa physical layer. In turn, the gateways transmit this data to a network server. This communication can be conducted through various technologies such as IEEE 802.3 (Ethernet), IEEE 802.11 (Wi-Fi), satellite, etc. [105]. The network server is responsible for controlling the data by executing the appropriate security operations and checking for redundant packets. Finally, it transmits the data to application servers that constitute software applications.

LoRaWan technology includes two security layers. The first security layer undertakes to authenticate the end nodes data. This process is carried out through an AES-CTR 128 secret key, called Network Session Key (NwkSKey) between the end nodes and the network server. On the other side, the second layer is responsible for assuring the privacy protection of end nodes by utilizing an AES-CTR 128 secret key called Application Key (AppSKey) between the end nodes and the application servers. Consequently, a crucial issue for the LoRaWAN technology is the safety of these keys. If any of them is stolen, then a potential attacker will be able to access and modify the specific data. Furthermore, concerning the communication between the end nodes and the gateways, it is worth mentioning that the payload length remains unchanged before and after the encryption. An attacker can exploit this situation, attempting to restore NwkSKey from the encrypted messages [106]. Moreover, an attacker with physical access to the end nodes has the ability to extract the aforementioned keys. More specifically, an end node includes a LoRa radio module and an MCU. The LoRa radio module interacts with MCU utilizing Universal Asynchronous Receiver Transmitter (UART) and Serial Peripheral Interface (SPI) interfaces. However, LoRa radio module does not include embedded encryption mechanisms, thus enabling the attacker to extract the keys, by using external hardware equipment such as a Future Technology Devices International (FTDI) interface [106]. In [106], the authors demonstrate this vulnerability by utilizing physical equipment. Additionally, it should be noted that the LoRaWAN packets do not integrate time information to verify the integrity of the messages. This situation could be

lead to replay and wormhole attacks. In [106], the authors describe the process of a possible wormhole attack against LoRaWAN. Finally, in [107], B.Reynders et al. demonstrate that the LoRa transmissions are prone to jamming attacks. Specifically, simultaneous communications that employ the same spreading factor and frequency can conflict with each other. E.Aras et al. in [106] present how an attacker can use Commercial-Off-The-Shelf (COTS) LoRa devices to perform a jamming attack against LoRa networks.

Concerning the possible security solutions for the aforementioned LoRaWAN security issues, the key management schemes can provide reliable solutions for the safety of the encryption keys. In particular, most of the key management schemes in the IoT applications utilize cryptographic algorithms, such as Diffie Hellman and Elliptic Curve Cryptography (ECC) [105]. More specifically, in [108] the authors present a key management scheme which generates a session key for two different entities, by combing the Elliptic Curve Qu-Vanstone (ECQV), ECDH and nonces. ECQV is used to provide appropriate certificates. Accordingly, ECDH is responsible for the data exchange and finally, the nonces guarantee the protection from replay attacks. Moreover, in [109] the authors provide a key management scheme for the IoT devices, utilizing intermediate nodes called proxies. The IoT devices have the ability to establish a shared session key, while the proxies undertake the complex cryptographic processes. In [105] S.Naoui et al. improve the functionality of the previous scheme in [109], by providing a reputation mechanism which offers the capability to opt trustworthy nodes. Finally, regarding the jamming, replay and wormhole attacks, thereafter of the article, we provide appropriate countermeasures such as anti-jamming solutions and IDPS systems.

7.2.6. 6LoWPAN Security

Utilizing the IEEE 802.15.4 protocol at the PHY and the MAC sublayers, the Low Power Wireless Personal Area Networks (WPANs) can use only 102 bytes for the transmission of information at next communication layers. However, the value of the Maximum Transmission Unit (MTU) that is needed for the IPv6 requirements is equivalent to 1280 bytes which is considerably higher than the previous number. The purpose of the IPv6 low power WPAN (6LoWPAN) standard is to solve this complication by deploying the interconnection between the IEEE 802.15.4 and IPv6 protocols for WPANs. In particular, it operates as an adaptation layer that utilizes compression, fragmentation and encapsulation mechanisms and transmits the modified IPv6 packets at the MAC sublayer.

Currently, 6LoWPAN standard does not provide any security mechanism, such as IPSec due to the limitations of IoT devices [15, 110]. However, individual research proposals [111–114] examine possible solutions to address these constraints, designing compressed security headers for the 6LoWPAN adaptation layer which have the same purpose as the existing Encapsulating Security Payload (ESP) and Authentication Header (AH) of IPSec. Also, some studies [115, 116] consider the incorporation of specific mechanisms in the 6LoWPAN against fragmentation attacks. More specifically, the authors in [115] discuss the addendum of a timestamp and a nonce field to the 6LoWPAN fragmentation header in order to address such attacks. In addition, [116] proposes the use of mechanisms that can support the pre-fragment sender authentication and prevent messages that are considered as suspicious. Finally, a significant security addition to the 6LoWPAN standard is the key management,

as the keys must be regularly renewed in order to assure the principles of confidentiality, integrity and authenticity. For instance, the Internet Key Exchange version 2 (IKEv2) protocol could be adopted, which is appropriate for use in devices with constrained resources. Therefore, as a result, the lack of security mechanisms in the 6LoWPAN standard offer research opportunities for improvements in future versions.

7.2.7. RPL Security

The RPL protocol was created by the Internet Engineering Task Force (IETF) and is appropriate to route messages in Low Power and Lossy Networks (LLNs). Its operation is based on the creation of a Destination Oriented Directed Acyclic Graph (DODAG) that utilizes an objective function [42]. In more detail, the DODAG consists of a set of nodes, which possess oriented edges in order not to create loops. The creation of a DODAG starts when the root node transmits a DIO message to their neighbors. The neighboring nodes receive the DIO message and take the decision whether they join in the graph. If a node joins the graph, then the corresponding path to the root node is created. Then, using the objective function, the new node of the graph calculates a value which is called rank. This procedure is repeated for each node in the graph. Finally, it is worth mentioning that the nodes have the ability to transmit a DODAG Information Solicitation (DIS) message in order to discover new DODAGs and as well as they can send DODAG Destination Advertisement Object (DAO) messages to advertise a routing path.

The security in the RPL protocol is based on the existence of secure variations of the RPL packets (DIS, DIO, DAO, DAO-ACK) and also the capability to apply three security modes. These variations provide integrity, replay protection, delay protection and optional confidentiality. Specifically, the cryptographic algorithms and the overall security strategy are identified by the Security field that is analyzed further in the following subfields [117].

- **Counter is Time (T)**: If this field possesses a value, then it represents a timestamp. On the other hand, it is employed as a counter.
- **Reserved**: The transmitter should initialize the specific field to zero and the receiver have to disregard it.
- **Algorithm**: The specific field determines basic features of RPL secured packets, such as the encryption algorithm, the signature algorithm and the message authentication code. In particular, the supported options are AES-CCM 128-bit and RivestShamirAdleman (RSA) combined with the Secure Hash Algorithm (SHA)-256 hash function.
- **Key Identifier Mode (KIM)**: This field consists of two bits that identify if the encryption key has defined indirectly or directly and also indicate the form of Key Identifier field respectively. More precisely, it can support group keys, per-pair keys and signature keys.
- **Resvd**: The transmitter should initialize the specific field to zero and the receiver has to disregard it.

- **Security Level (LVL)**: The specific field consists of three bits that determine the security of the packet. According to KIM field, LVL can provide different levels of data authenticity and confidentiality.
- **Flags**: The transmitter should initialize the specific field to zero and the receiver have to disregard it.
- **Counter**: This field involves the required information for the creation of the cryptographic mechanism.
- **Key Identifier**: This field identifies the encryption keys. It supports multiple keying models, such as group keys, peer-to-peer keys and signature keys. More specifically, it is separated into Key Source and Key Index subfields. The Key Source subfield occupies 8 bytes and it is employed in order to identify the group key originator. On the other hand, the size of the Key Index subfield is 1 byte and it is utilized to recognize keys that are controlled by the same originator.

Previously, we described how the security-related information is incorporated in the RPL control messages. In this paragraph, we will discuss the three security modes, which can be employed by the RPL protocol. The default usage mode is the "unsecured", in which the RPL messages are transmitted without any additional security mechanism. In the second mode, called "preinstalled", the nodes must possess a pre-configured key, which is employed to connect to the DODAG as a router or a host. Finally, the third mode, called "authenticated" is appropriate for the nodes that operate as routers. Specifically, the nodes must have a pre-configured key as the previous case, and then they must acquire a different key from a validated authority.

In conclusion, the RPL protocol provides significant security mechanisms that guarantee the confidentiality, the integrity and the authenticity of information. However, as mentioned above the addressing of the routing attacks is still an important security issue. Research efforts have to be focused on appropriate mechanisms that can address these threats. Such mechanisms are discussed in the subsection 7.2.13.

7.2.8. TLS Security

Many IoT application protocols, such as Message Queuing Telemetry Transport (MQTT), Extensible Messaging and Presence Protocol (XMPP), and Advanced Message Queuing Protocol (AMQP) utilize TLS in order to guarantee security in the application layer communications. Specifically, the TLS protocol deploys security mechanisms at the transport sublayer and consists of individual protocols which are separated into two layers. The first layer includes the Record Protocol while the second layer contains the Alert Protocol, the Change Cipher Spec Protocol, the Handshake Protocol and the Heartbeat Protocol.

Regarding the operation of the Record Protocol, the application data are separated into blocks of 214 bytes or less. Subsequently, these blocks of data are compressed optionally. Next, a message authentication code (mac) is computed for the specific blocks and a symmetric encryption algorithm is employed in order to encrypt them and the mac. The final

step is the addition of a specific header which includes version and length fields. After the aforementioned procedure, the Record Protocol encapsulates the combined total information in a Transmission Control Protocol (TCP) packet and transmits it. On the other hand, the received information is decrypted, validated, reassembled and then delivered to higher-layer endpoints.

The operation of the Change Cipher Spec Protocol is based on a single byte, which always has the value 1 and copies the pending state to the current state in order to update the cryptographic algorithm and its characteristics, which will be employed for the specific connection.

The Alert Protocol provides alerts about the overall TLS operation. Each message of this protocol involves two bytes. The functionality of the first byte is to indicate the severity level of the message. In particular, it can obtain the values (1) warning or (2) fatal. In the case that the severity level is fatal, then the TLS directly aborts the specific connection. The remaining connections of the session can exist; however, any new connection cannot be created. On the other hand, the second byte comprises a code, which identifies the particular alert.

The Handshake Protocol implements the most critical operations of the TLS, and it is applied before any application data transmission. More precisely, it includes two primary operations: (1) an authentication process for the server and the client and (2) a negotiation process of the encryption algorithm, the mac and the cryptographic keys. The overall operation of the Handshake Protocol can be divided into four phases. In phase 1, the client starts the communication by transmitting a ClientHello message that includes the following information:

- **Version**: This field implies the last version of the TLS protocol, which is utilized by the client.
- **Random data**: This field indicates a data structure which consists of a timestamp that occupies 32 bit and a random value that involves 28 bytes.
- **Session ID**: This field refers to the session identifier. If the value of the specific field is equal to zero that means that the client desires to renew the features of a current connection or to establish a different connection for the particular session. On the other hand, a value equal to zero implies that the client needs to create a new connection for a different season.
- **CipherSuite**: The CipherSuite field identifies the combinations of the cryptographic algorithms that can be employed by the client.
- **Compression methods**: This field carries the compression methods that can be utilized by the client.

When the server receives the ClientHello message, it transmits to the client a ServerHello message that contains the similar types of information such as the ClientHello message.

The operations of phase 2 are determined according to the requirements of the asymmetric encryption algorithm which will be employed. Usually, the server sends information about the public-key encryption process, such as the certificate, details about the cryptographic keys as well as a request for a certificate from the client. The last work of the particular procedure is always the transmission of a ServerDone message, which indicates the end of this phase.

Similarly, the processes of phase 3 are defined according to the characteristics of the asymmetric cryptographic algorithm. Initially, the client validates the server's certificate and checks the information that is included in the messages of phase 2 and in the ServerHello message of phase 1. Subsequently, it sends back to the server one or more messages, which are associated with the public-key encryption scheme. For instance, it can send a certificate if requested, key exchange information, and certificate verification.

Phase 4 completes the establishment of the secure connection, which implements the Handshake Protocol. In particular, the client transmits a ChangeCipherSpec message and renews the pending CipherSpec in the current CipherSpec. Next, it sends to the server a finished message which indicates that the verification of authentication and key exchange processes was succeeded swimmingly. On the other hand, the server transmits to the client its own ChangeCipherSpec message, refreshes the pending to the current CipherSpec, and finally transmits its finished message.

The Heartbeat protocol serves two purposes. First, it assures the sender that the receiver is still alive, in a specific TCP connection. Secondly, it creates additional network activity in idle connections in order to avoid the closure of these connections by a firewall or an intrusion detection and prevention system. More specifically, the Heartbeat protocol involves two kinds of messages: a HeartbeatRequest packet and a HeartbeatResponse message. A HeartbeatRequest message can be used at any time. Whenever a HeartbeatRequest message is received, it has to be answered immediately with a corresponding eartbeatResponse message. It is noteworthy that the requirement of the Heartbeat protocol was introduced for the DTLS protocol. However, for reasons of simplicity, the same version of the Heartbeat protocol is utilized with both TLS and DTLS.

TLS performs significant security operations which assure the principles of confidentiality, authenticity and integrity of communications; nevertheless, it is an expensive protocol to use in IoT devices [118].

7.2.9. DTLS Security

The DTLS protocol is a variation of the TLS and assures the existence of the same security principles. In particular, DTLS operates over datagrams which can be lost, duplicated or received in wrong order. For this reason, in comparison with the TLS, it supports some additional mechanisms which are listed below. The RFC 6347 document [119] explains these mechanisms in great detail.

- One difference is the extension of the TLS Record Protocol with two additional fields. In particular, an epoch and a sequence number field are utilized in order to compute the mac.

- The DTLS does not permit the utilization of the stream ciphers.
- The operation of the TLS Handshake Protocol is improved with the addition of a stateless cookie and also it is able to address fragmentation, message loss and reordering. More specifically, the phase 1 of the TLS Handshake Protocol is differentiated as follows: The client transmits its ClientHello message. The server responds to the client with a HelloVerifyRequest packet which contains a stateless cookie, that must be transmitted back as a second ClientHello message.

Many IoT application protocols such as the MQTT for Sensor Networks (MQTT-SN) and the CoAP utilize the DTLS; however, the DTLS is characterized by several limitations. More specifically, the DTLS Handshake Protocol may not probably support some IoT communications, as the large messages can be fragmented at the 6LoWPAN adaptation layer [15]. This state may lead to retransmission of some packets, which may produce complications. Also, the calculation and transmission procedures of the finished messages are costly for the IoT devices [15]. In addition, the DTLS cannot be efficiently applied to some application protocols. For instance, the DTLS is not suitable to be used in CoAP proxies [15]. Furthermore, the future IoT applications may need to incorporate the online verification of X.509 certificates; the mechanisms which implement this functionality require further investigation [15]. Finally, the DTLS cannot implement multicast communications [15]. Therefore, in conclusion, research efforts could be focused on appropriate mechanisms that can address these limitations.

7.2.10. CoAP Security

The CoAP protocol can be considered as a lightweight version of the HTTP protocol which is devoted to implementing the communications at the application layer for power constrained IoT devices. It runs over the User Datagram Protocol (UDP) utilizing 6LoWPAN and follows the Representational State Transfer (RES) architectural scheme which consists of the methods: (1) GET, (2) POST, (3) PUT and (4) DELETE. The architecture of CoAP consists of two layers: (1) message layer and (2) request/response layer. The message layer undertakes to control the communication over the UDP protocol, while, the request/response protocol is responsible for sending the corresponding messages, by maintaining specific codes that are used to manage and avoid functional issues such as the loss of messages [120, 121].

The security of CoAP is mainly based on the adoption of the DTLS protocol at the transport layer. Usually, the implementation of DTLS over CoAP is referred to as CoAPs. As mentioned before, DTLS assures the confidentiality, integrity and non-repudiation of information and services. The security provided by DTLS, adds 13 bytes overhead per datagram. More detailed, CoAP involves four security modes: (1) NoSec, (2) PreSharedKey, (3) RawPublicKey and (4) Certificates [15, 120]. The first mode does not incorporate any security mechanism. The second mode utilizes symmetric encryption, where each device possesses pre-programmed symmetric keys. The applications that utilize the specific method have to support the TLS_PSK_WITH_AES_128_CCM_8 [122] suite. The RawPablicKey mode is devoted to establishing asymmetric encryption on such devices that

cannot use the Public Key Infrastructure (PKI) technology. Each device possesses a pre-programmed pair of private and public keys that are determined during the manufacturing process. The public key constitutes the identity of the corresponding device. Furthermore, each device possesses a catalog of public keys which indicates the other devices with which it can communicate. The applications use the particular mode have to support the TLS_ECDH_ECDS_WITH_AES_128_CCM_8 suite [123, 124]. Finally, the last mode requires the presence of a trusted authority thus making possible the functionality of PKI. Each device is identified by an X.509 certificate. The applications that use this method should apply either the TLS_ECDH_ECDS_WITH_AES_128_CCM_8 suite [123, 124] or the TLS_ECDHE_PSK_WITH_AES_128_CBC_SHA suit. Adopting DTLS, it is noteworthy that the security processes of the two last modes (RawPublicKey and Certificates) are implemented through ECC [15, 120]. Specifically, the Elliptic Curve Digital Signature (ECDS) algorithm is responsible for the device authentication, while the ECDH counterpart and ECDH with Ephemeral Keys (ECDHE) undertake the key agreement.

According to the above, the security of CoAP depends on DTLS. However, as described in the previous subsection, DTLS suffers from various functional issues in an IoT environment. Consequently, maybe CoAP should integrate new security solutions without utilizing DTLS. In [125], the authors introduce three new security options for CoAP. Specifically, the first option identifies the security capabilities of a CoAP packet as well as the security measures and requirements of the entity which is responsible for securing this packet. The second option determines how the respective data can be transmitted for the authentication and authorization processes. Finally, the third option is responsible for determining how the data should be transmitted for the encryption processes.

7.2.11. Protection against Jamming Attacks

None of the previous protocols is able to provide efficient countermeasures against the jamming attacks. This subsection aims at providing a general overview of possible solutions against the specific kind of attacks. In particular, these solutions can be classified into four categories: (1) detection techniques, (2) proactive countermeasures, (3) reactive countermeasures, and (4) mobile agent-based countermeasures. The countermeasures of the first category aim to detect the jamming attacks instantly, but they can offer notable protection only if they are coupled with the other models. On the other hand, the proactive countermeasures pursue to render an IoT system resistant to jamming attacks. More precisely, they can effectively address the models of constant jamming, deceptive jamming, and random jamming. Also, they can be classified further into software and combined software-hardware solutions. The third protection model takes into consideration the energy consumption and is an appropriate solution for the reactive jamming attacks. In more detail, it operates only if the IoT devices perceive a jamming attack. Also, as the previous model, it can be categorized into software and coupled software-hardware solutions. Finally, the last model utilizes mobile agents, which is able to be transferred to multiple IoT devices and attempt to abort the jamming attacks. However, this model is characterized by high implementation cost and complexity. Various countermeasures against jamming attacks are listed in [126–130]. In [126] the authors provide a game theoretic approach to prevent jamming attacks.

Specifically, this method is based on the Colonel Blotto game where a controller device can hinder jamming attacks against the IoT devices. Similarly, in [127] the authors implement a hierarchical game which mitigates the jamming attacks. Another Colonel Boloto game against jamming attacks is presented in [128]. Accordingly, in [129] H. A. Bany Salameh et al. pay their attention to proactive and reactive jamming attacks, providing a probabilistic method which reduces the ratio of invalid cognitive radio transmissions. Finally, in [130] Y.Chen et al. provide a deep reinforcement learning model which is devoted to saving power and optimizing the transmission performance, thus mitigating the jamming a

7.2.12. Firewalls

A firewall is a protection system in the form of hardware or software which continuously controls the network activities by using a set of predefined rules. These systems have the ability to monitor the network traffic at a number of levels, from low-level network packets to application protocols packets. The choice of the level is determined by the desired firewall access policy, which should, in turn, be defined by the security management and risk assessment processes. More specifically, the firewalls can be categorized either by their operation mode or by their placement. In the first case, the firewall can be classified into four categories: (1) Packet filtering firewall, (2) Stateful inspection firewall, (3) Application-level gateway and (4) Circuit-level gateway [74]. On the other hand, different topologies can be created depending on the risk assessment processes. A firewall in an IoT environment can be installed either in the IoT devices or in a central intermediate node which will be responsible for the communication between the IoT devices and the conventional ICT systems. Some firewalls for the IoT are listed in [131, 132]. In particular, in [131] the authors present a centralized firewall which operates in the context of an IDPS system, proving appropriate rules based on the detected attacks. Accordingly, in [132], N. Gupta et al. provide a firewall for IoT which utilizes a Raspberry Pi device as a gateway as well as a cloud database, adopting heuristic functions and signature rules.

7.2.13. Intrusion Detection and Prevention Systems

The IDPS systems comprise a set of protection mechanisms that aim to detect, record and prevent potential threats in real-time. They can control information, which is generated by multiple computing resources, such as the system calls of OS or the network activities. More specifically, as in the case of firewalls, these systems can be classified either by their operation mode or by their placement. In the first case, three types of IDPS are distinguished: (1) Signature-based IDPS, (2) Anomaly-based IDPS and (3) Specification-based IDPS. he operation of the signature-based IDPS is based on the comparison of the monitored data with a set of predetermined threat models called signatures. This approach presents high accuracy rate, but cannot counter new types of threats and is characterized by a significant storage cost. The anomaly-based IDPS attempt to identify possible anomalies by using statistical models or machine learning techniques, such as Bayesian networks, genetic algorithms and Markov chains. This method has the advantage to prevent new types of threats, but typically presents high false positive rate and is characterized by important computation cost. Finally, the third method analyzes the monitored data with a set of rules

that determine the normal function of the system. As in the previous case, this approach can detect unknown threats but in a dynamic environment such as the IoT, these rules may change continuously or periodically. On the other side, as in the case of the firewall systems, an IDPS can be installed either in the IoT nodes or in an intermediate node. Some IDPS for the IoT are listed in [21, 131, 133–135]. In [131] the authors provide an IDPS system for 6LoWPAN-RPL networks which is able to detect sinkhole, sybil and selective forwarding attacks utilizing a hybrid approach which combine signatures an anomaly-based method. In [133] M. Surendar and A. Umamakeswari provide an IDPS which devoted to detecting sinkhole attacks adopting particular specifications. D. Midi et al. in [134] propose Kalis which constitutes an IDPS capable of monitoring and controlling multiple communication protocols, combining signature rules and anomaly detection processes. Finally, C. Cervantes et al. in [135] presents INTI which performs reputation and trust mechanisms to detect sinkhole attacks. An analytical survey of IDPS systems in the context of IoT is listed in [21].

7.3. Countermeasures at the Support Layer

The security mechanisms at the support layer have to control the unauthorized access, the malicious insiders, the insecure software interfaces, and the unknown risk profile threats. First, it should be ensured that only legitimate users and objects are able to utilize the services and the data of the storing systems. Therefore, remote authentication systems, access control mechanisms and a trust framework are required. Also, secure programming techniques, firewalls and IDPS systems are important countermeasures as they can prevent the data loss or leakage. Finally, concerning the malicious insiders, possible security solutions include the implementation of a specific policy which involves stringent management and security rules, the specification of notification processes and the requirement of transparency into the overall information security and management processes [74].

7.4. Countermeasures at the Business Layer

The security mechanisms at the business layer have to ensure the protection of the software applications and the Operating System (OS) of the IoT devices and the user interfaces. The main security threats of the software applications are due to the insecure programming. Possible security solutions for this issue include the utilization of a high-level programming language, which automatically manages the memory issues. Some examples are Java, Python and C#. Also, as in the support layer, the OS security should be enhanced by various security tools and processes, such as access control and IDPS systems. A survey paper which analyzes and evaluates various authentication and access control systems for IoT is listed in [20]. Finally, management and education processes are very critical at this level, as they can protect the users from social engineering techniques.

8. Discussion and Research Directions

In this paper, we aimed at providing a comprehensive analysis regarding the security issues in IoT. Our goal was to provide a study which lists and evaluates the potential security threats in IoT, identifies the possible vulnerabilities and limitations of the IoT protocols

and finally proposes appropriate solutions. Firstly, we introduced the necessary security requirements as well as the various challenges that in contrast to the traditional computing systems can difficult and limit the efforts for deploying appropriate security countermeasures. Next, by introducing a risk assessment model, we conducted a risk assessment, evaluating qualitatively and quantitively the various security threats in IoT. Subsequently, we discussed the possible countermeasures per layer. Specifically, in this section, we mainly focused on the communication protocols, by analyzing and identifying their security features and vulnerabilities.

Based on our risk assessment, the security threats at the perception layer cannot be considered as very crucial, as the risk level of the natural disasters and environmental threats is low, while the risk of the human-caused physical attacks is medium. In particular, appropriately designed infrastructures, as well as physical authentication and access control systems, are sufficient to prevent these threats. Nevertheless, it should be noted that the impact of these threats can be disastrous making possible tremendous health, economic and social consequences. For instance, in the case of the smart grid, a physical or a cyberattack can generate severe issues even human losses. Therefore, we consider that the research efforts in this field should focus on self-healing mechanisms, thus protecting critical locations and systems. Specifically, based on the kind of emergency, the self-healing mechanisms should be able to either isolate critical systems and locations, thus ensuring their protection or activate recovery and collaborating mechanisms to mitigate the possible attacks. Of course, these mechanisms should take into account both of physical and cyber threats.

Based on our risk assessment model, the most dangerous threats in the communication layer are the jamming and MiTM attacks. The jamming attacks threaten all the PHY and MAC communication protocols, while the MiTM attacks remain a critical threat for all protocols, despite the existing encryption mechanisms. For instance, ZigBee, BLE, LoRaWan and in some cases (e.g., malicious proxies) the application protocols (e.g., CoAP) suffer from replay attacks. The routing attacks, such as selective forwarding attacks, sybil attacks, Hello Flood, sinkholes and wormholes are characterized as a medium threat, since the routing protocols (e.g., RPL) and IDPS systems possess mechanisms to address them. Finally, although the impact of the traffic analysis attacks is moderate, they are considered as a medium threat, since they can constitute the first step for other cyberattacks. Regarding the jamming attacks, the game-theoretic approaches seem to be a promising solution to model and address this kind of threats. Consequently, the research efforts can pay their attention to such methods. According to the literature [126–128], hierarchical and Colonel Bloto games can be adopted to hinder these attacks. Concerning, the MiTM attacks, a crucial security issue which affects the safety of all communication protocols is the key management. Hence, the research efforts can focus to implement appropriate key management protocols for the IoT communications. Another research direction capable of addressing the MiTM and routing attacks is the firewall and IDPS systems. Although firewall constitutes a crucial system for the overall security of an IoT network, we found only a few works in this area [131, 132]. A firewall for the IoT should be able to be deployed in a distributed manner without reducing the performance of the IoT network. It should include multiple agents that will be able to inspect all the communication layers and identify possible critical states

timely. Concerning the IDPS systems, it is clear that their presence is necessary for the security of the IoT networks. As in the case of the firewall systems, their function should not hinder the efficiency of the IoT devices. Most of the current IDPS systems are characterized by functional issues, since they affect either the storage capacity or the energy power of the IoT devices. We consider that the specification-based IDPS that are deployed and devoted to specific applications can overtake these issues. The Software Defined Networking (SDN) technology can contribute significantly to this direction [136, 137]. In particular, SDN provides virtualization and global visibility capabilities, making possible the generation of the specification rules. Moreover, SDN enables the splitting of a network into individual parts, thus making it easier to control the overall network.

At the support layer, the most critical threats are the malicious insiders and the unauthorized access. Both of them can result in catastrophic consequences. An effective solution against these attacks is the application of a stringent access control framework. In particular, the specific mechanism can be defined as the process which grants or denies particular requests to obtain and use information or related computing services. In more detail, it consists of three functions: (1) authentication of users or system entities, (2) authorization which determines the access privileges and (3) audit, which constitutes an independent and continuous analysis of system activities to ensure the adequacy of the previous functions. Based on these functions, the access control mechanisms can be classified into four categories: (1) Discretionary Access Control (DAC), (2) Mandatory Access Control (MAC), (3) Role-Based Access Control (RBAC) and (4) Attribute-Based Access Control (ABAC). Among these models, the most used is RBAC which assigns specific roles to the data requestors with corresponding access permissions. However, we consider that this model is inappropriate to identify the access privileges of the IoT devices because they can change roles continually depending on their function. On the other side, the ABAC model is more flexible, possessing the capability to identify the access control rules depending on the attributes of each device. Consequently, future research works can focus on the ABAC access control models, taking into consideration the constraints of IoT. In [23], the authors provide a comprehensive analysis and research directions regarding the access control systems in IoT. A promising approach in this domain is the combination of the ABAC model with the blockchain technology [138–142]. Specifically, the smart contracts technology of the Ethereum blockchain [143] could be used to identify the potential access control rules, thus including the benefits of the blockchain technology.

At the business layer, the most critical threat is the social engineering techniques, while the buffer overflow attacks and backdoors are less dangerous since the can be addressed with secure programming techniques. No countermeasure can prevent the social engineering attacks; nevertheless, anomaly detection and recovery mechanisms can mitigate them. Consequently, the presence of the IDPS systems and the self-healing mechanisms is also necessary at this level.

The various kinds of DoS attacks, rootkits and APTs present a high-risk level for all layers. These threats cannot fully be prevented, especially in the case of APTs, where the attackers have studied very carefully their actions. On the other side, the cryptanalytic attacks, botnets and spyware constitute threats of a medium level, since encryption

mechanisms and antivirus software may be able to address them. The interconnected and independent nature of IoT devices, as well as the heterogeneity of technologies, can generate multiple unknown vulnerabilities that can be exploited by various attackers. Furthermore, the hacking toolkits have largely been evolved where even a novice is able to achieve devastating consequences. These situations demand the development of appropriate tools that can monitor and control the IoT devices in a large scale manner. The System Information and Event Management (SIEM) tools can provide this capability by integrating aggregation, normalization and correlation services [144–146]. Usually, these tools consist of multiple software packages that are responsible for various services, such as data collection, intrusion detection, availability checking and vulnerability scanning. For this reason, they were not included in the "Countermeasures" section. However, these systems have not yet been adapted to the world of IoT, not having the ability to process protocols. Therefore, adapting these systems or creating new ones in the I environment is a major challenge.

9. Conclusions

The IoT is an emerging technology that has significantly attracted the interest both of industry and academia. Despite the fact that it is still at an early age of development, several applications have been implemented, such as smart grid, smart home and smart city. The use and evolution of this technology depend mainly on the security aspect, since now the security measures should control the actions both of users and objects. At the same time, IoT generates new security challenges, since the IoT devices are characterized by constrained computing resources, making impossible the use of the conventional security mechanisms. Furthermore, the multiple interconnections and heterogeneity of devices produce huge amounts of data that are difficult to manage. Therefore, the security analysis in IoT is a critical need in order to take the appropriate measures.

In this paper, we conducted a comprehensive security analysis of IoT, focusing our attention on the possible threats and the corresponding countermeasures. Once we identified the security requirements and challenges, we evaluate the possible threats by introducing a risk assessment model. For each threat, we determined its risk level utilizing three values: (1) Low, (2) Medium and (3) High. These values are characterized by specific quantitative limits. Next, we determined the corresponding countermeasures, paying our attention to the security mechanisms of the IoT protocol. The protocols we examined are: IEEE 802.15.4, ZigBee, Z-Wave, BLE, LoRaWan, 6LoWPAN, RPL, TLS, DTLS and CoAP. For each protocol, we analyzed its security features and identified potential vulnerabilities and limitations. Based on this analysis, we provide possible directions for future research work.

In our future work, we intend to exploit the benefits of the SDN technology in order to provide an efficient hybrid IDPS system for specific IoT applications. The proposed IDPS will be based on a cross-layer model of IoT [147] and combine anomaly detection techniques with specification rules, thus detecting with high accuracy the possible cyberattacks. Furthermore, appropriate visualization mechanisms will enhance the functionality of IDPS, identifying possible cyberattacks patterns. Finally, we will study the integration of

the proposed system in a SIEM tool, such as AlienVault's OSSIM [148] in order to optimize its capabilities and prevent timely possible attacks.

References

References

[1] K. Ashton, et al., That internet of things thing, RFID journal 22 (7) (2009) 97–114.
[2] J. B. Kennedy, When woman is boss: an interview with nikola tesla, {http://www.tfcbooks.com/tesla/1926-01-30.htm} (1926).
[3] L. Atzori, A. Iera, G. Morabito, Understanding the internet of things: definition, potentials, and societal role of a fast evolving paradigm, Ad Hoc Networks 56 (2017) 122–140. doi:https://doi.org/10.1016/j.adhoc.2016.12.004.
URL http://www.sciencedirect.com/science/article/pii/S1570870516303316
[4] A. M. Turing, Computing machinery and intelligence., Creative Computing 6 (1) (1980) 44–53.
[5] A. M. Turing, Computing machinery and intelligence, in: Parsing the Turing Test, Springer, 2009, pp. 23–65.
[6] J. Rivera, R. van der Meulen, Gartner says the internet of things installed base will grow to 26 billion units by 2020, {https://www.gartner.com/newsroom/id/2636073} (2013).
[7] C. Cheng, R. Lu, A. Petzoldt, T. Takagi, Securing the internet of things in a quantum world, IEEE Communications Magazine 55 (2) (2017) 116–120. doi:10.1109/MCOM.2017.1600522CM.
[8] H. Suo, J. Wan, C. Zou, J. Liu, Security in the internet of things: A review, in: 2012 International Conference on Computer Science and Electronics Engineering, Vol. 3, 2012, pp. 648–651. doi:10.1109/ICCSEE.2012.373.
[9] S. A. Kumar, T. Vealey, H. Srivastava, Security in internet of things: Challenges, solutions and future directions, in: 2016 49th Hawaii International Conference on System Sciences (HICSS), 2016, pp. 5772–5781. doi:10.1109/HICSS.2016.714.
[10] P. Schaumont, Security in the internet of things: A challenge of scale, in: Design, Automation Test in Europe Conference Exhibition (DATE), 2017, 2017, pp. 674–679. doi:10.23919/DATE.2017.7927075.
[11] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, W. Zhao, A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications, IEEE Internet of Things Journal 4 (5) (2017) 1125–1142. doi:10.1109/JIOT.2017.2683200.
[12] I. Andrea, C. Chrysostomou, G. Hadjichristofi, Internet of things: Security vulnerabilities and challenges, in: 2015 IEEE Symposium on Computers and Communication (ISCC), 2015, pp. 180–187. doi:10.1109/ISCC.2015.7405513.
[13] E. Leloglu, A review of security concerns in internet of things, Journal of Computer and Communications 5 (1) (2017) 121–136. doi:10.4236/jcc.2017.51010.
[14] A. Mosenia, N. K. Jha, A comprehensive study of security of internet-of-things, IEEE Transactions on Emerging Topics in Computing 5 (4) (2017) 586–602. doi:10.1109/TETC.2016.2606384.
[15] J. Granjal, E. Monteiro, J. S. Silva, Security for the internet of things: A survey of existing protocols and open research issues, IEEE Communications Surveys Tutorials 17 (3) (2015) 1294–1312. doi:10.1109/COMST.2015.2388550.
[16] K. T. Nguyen, M. Laurent, N. Oualha, Survey on secure communication protocols for the internet of things, Ad Hoc Networks 32 (2015) 17 – 31, internet of Things security and privacy: design methods and optimization. doi:https://doi.org/10.1016/j.adhoc.2015.01.006.
URL http://www.sciencedirect.com/science/article/pii/S1570870515000141
[17] R. Krej, O. Hujk, M. vepe, Security survey of the iot wireless protocols, in: 2017 25th Telecommunication Forum (TELFOR), 2017, pp. 1–4. doi:10.1109/TELFOR.2017.8249286.
[18] D. Celebucki, M. A. Lin, S. Graham, A security evaluation of popular internet of things protocols for manufacturers, in: 2018 IEEE International Conference on Consumer Electronics (ICCE), 2018, pp. 1–6. doi:10.1109/ICCE.2018.8326099.

[19] M. Sain, Y. J. Kang, H. J. Lee, Survey on security in internet of things: State of the art and challenges, in: 2017 19th International Conference on Advanced Communication Technology (ICACT), 2017, pp. 699–704. doi:10.23919/ICACT.2017.7890183.

[20] S. Sicari, A. Rizzardi, L. Grieco, A. Coen-Porisini, Security, privacy and trust in internet of things: The road ahead, Computer Networks 76 (2015) 146 – 164. doi:https://doi.org/10.1016/j.comnet.2014.11.008.
URL http://www.sciencedirect.com/science/article/pii/S1389128614003971

[21] B. B. Zarpelo, R. S. Miani, C. T. Kawakani, S. C. de Alvarenga, A survey of intrusion detection in internet of things, Journal of Network and Computer Applications 84 (2017) 25 – 37. doi:https://doi.org/10.1016/j.jnca.2017.02.009.
URL http://www.sciencedirect.com/science/article/pii/S1084804517300802

[22] M. Ammar, G. Russello, B. Crispo, Internet of things: A survey on the security of iot frameworks, Journal of Information Security and Applications 38 (2018) 8 – 27. doi:https://doi.org/10.1016/j.jisa.2017.11.002.
URL http://www.sciencedirect.com/science/article/pii/S2214212617302934

[23] A. Ouaddah, H. Mousannif, A. A. Elkalam, A. A. Ouahman, Access control in the internet of things: Big challenges and new opportunities, Computer Networks 112 (2017) 237 – 262. doi:https://doi.org/10.1016/j.comnet.2016.11.007.
URL http://www.sciencedirect.com/science/article/pii/S1389128616303735

[24] Y. Yang, L. Wu, G. Yin, L. Li, H. Zhao, A survey on security and privacy issues in internet-of-things, IEEE Internet of Things Journal 4 (5) (2017) 1250–1258. doi:10.1109/JIOT.2017.2694844.

[25] L. Chen, S. Thombre, K. Jrvinen, E. S. Lohan, A. Alm-Savikko, H. Leppkoski, M. Z. H. Bhuiyan, S. Bu-Pasha, G. N. Ferrara, S. Honkala, J. Lindqvist, L. Ruotsalainen, P. Korpisaari, H. Kuusniemi, Robustness, security and privacy in location-based services for future iot: A survey, IEEE Access 5 (2017) 8956–8977. doi:10.1109/ACCESS.2017.2695525.

[26] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, M. Ayyash, Internet of things: A survey on enabling technologies, protocols, and applications, IEEE Communications Surveys Tutorials 17 (4) (2015) 2347–2376. doi:10.1109/COMST.2015.2444095.

[27] Aditya, M. Sharma, S. C. Gupta, An internet of things based smart surveillance and monitoring system using arduino, in: 2018 International Conference on Advances in Computing and Communication Engineering (ICACCE), 2018, pp. 428–433. doi:10.1109/ICACCE.2018.8441725.

[28] U. Isikdag, Internet of things: Single-board computers, in: Enhanced Building Information Models, Springer, 2015, pp. 43–53.

[29] B. Board, Beagle bone black (2014).

[30] G. Chu, N. Apthorpe, N. Feamster, Security and privacy analyses of internet of things childrens toys, IEEE Internet of Things Journal (2018) 1–1doi:10.1109/JIOT.2018.2866423.

[31] D. K. Rahmatullah, S. M. Nasution, F. Azmi, Implementation of low interaction web server honeypot using cubieboard, in: 2016 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), 2016, pp. 127–131. doi:10.1109/ICCEREC.2016.7814970.

[32] A. Khattab, Z. Jeddi, E. Amini, M. Bayoumi, Rfid security threats and basic solutions, in: RFID Security, Springer, 2017, pp. 27–41.

[33] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, et al., Tinyos: An operating system for sensor networks, in: Ambient intelligence, Springer, 2005, pp. 115–148.

[34] O. Hahm, E. Baccelli, H. Petersen, N. Tsiftes, Operating systems for low-end devices in the internet of things: A survey, IEEE Internet of Things Journal 3 (5) (2016) 720–734. doi:10.1109/JIOT.2015.2505901.

[35] J. McBride, B. Arief, J. Hernandez-Castro, Security analysis of contiki iot operating system, in: Proceedings of the 2018 International Conference on Embedded Wireless Systems and Networks, EWSN ’18, Junction Publishing, USA, 2018, pp. 278–283.
URL http://dl.acm.org/citation.cfm?id=3234847.3234913

[36] Y. B. Zikria, M. K. Afzal, F. Ishmanov, S. W. Kim, H. Yu, A survey on routing protocols supported by the contiki internet of things operating system, Future Generation Computer Systems 82 (2018) 200 – 219. doi:https://doi.org/10.1016/j.future.2017.12.045.
URL http://www.sciencedirect.com/science/article/pii/S0167739X17324299

[37] S. Bhatti, J. Carlson, H. Dai, J. Deng, J. Rose, A. Sheth, B. Shucker, C. Gruenwald, A. Torgerson, R. Han, Mantis os: An embedded multithreaded operating system for wireless micro sensor platforms, Mob. Netw. Appl. 10 (4) (2005) 563–579.
URL http://dl.acm.org/citation.cfm?id=1160162.1160178

[38] F. Guan, L. Peng, L. Perneel, M. Timmerman, Open source freertos as a case study in real-time operating system evolution, Journal of Systems and Software 118 (2016) 19 – 35. doi:https://doi.org/10.1016/j.jss.2016.04.063.
URL http://www.sciencedirect.com/science/article/pii/S0164121216300383

[39] T. B. Chandra, P. Verma, A. K. Dwivedi, Operating systems for internet of things: A comparative study, in: Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies, ICTCS '16, ACM, New York, NY, USA, 2016, pp. 47:1–47:6. doi:10.1145/2905055.2905105.
URL http://doi.acm.org/10.1145/2905055.2905105

[40] R. Mahmoud, T. Yousuf, F. Aloul, I. Zualkernan, Internet of things (iot) security: Current status, challenges and prospective measures, in: 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), 2015, pp. 336–341. doi:10.1109/ICITST.2015.7412116.

[41] J. McCarthy, O. Alexander, S. Edwards, D. Faatz, C. Peloquin, S. Symington, A. Thibault, J. Wiltberger, K. Viani, Situational awareness for electric utilities nist sp 1800-7 practice guide, NIST Special Publication 1800-7 (2017) 1–241.

[42] L. Wallgren, S. Raza, T. Voigt, Routing attacks and countermeasures in the rpl-based internet of things, International Journal of Distributed Sensor Networks 9 (8) (2013) 794326. arXiv:https://doi.org/10.1155/2013/794326, doi:10.1155/2013/794326.
URL https://doi.org/10.1155/2013/794326

[43] W. Xu, K. Ma, W. Trappe, Y. Zhang, Jamming sensor networks: attack and defense strategies, IEEE Network 20 (3) (2006) 41–47. doi:10.1109/MNET.2006.1637931.

[44] S. Vadlamani, B. Eksioglu, H. Medal, A. Nandi, Jamming attacks on wireless networks: A taxonomic survey, International Journal of Production Economics 172 (2016) 76 – 94. doi:https://doi.org/10.1016/j.ijpe.2015.11.008.
URL http://www.sciencedirect.com/science/article/pii/S092552731500451X

[45] L. K. Bysani, A. K. Turuk, A survey on selective forwarding attack in wireless sensor networks, in: 2011 International Conference on Devices and Communications (ICDeCom), 2011, pp. 1–5. doi:10.1109/ICDECOM.2011.5738547.

[46] A. Mayzaud, R. Badonnel, I. Chrisment, A Taxonomy of Attacks in RPL-based Internet of Things, International Journal of Network Security 18 (3) (2016) 459 – 473,.
URL https://hal.inria.fr/hal-01207859

[47] A. D. Wood, J. A. Stankovic, Denial of service in sensor networks, Computer 35 (10) (2002) 54–62. doi:10.1109/MC.2002.1039518.

[48] A. Mathew, J. S. Terence, A survey on various detection techniques of sinkhole attacks in wsn, in: 2017 International Conference on Communication and Signal Processing (ICCSP), 2017, pp. 1115–1119. doi:10.1109/ICCSP.2017.8286550.

[49] P. Nagrath, B. Gupta, Wormhole attacks in wireless adhoc networks and their counter measurements: A survey, in: 2011 3rd International Conference on Electronics Computer Technology, Vol. 6, 2011, pp. 245–250. doi:10.1109/ICECTECH.2011.5942091.

[50] K. Zhang, X. Liang, R. Lu, X. Shen, Sybil attacks and their defenses in the internet of things, IEEE Internet of Things Journal 1 (5) (2014) 372–383. doi:10.1109/JIOT.2014.2344013.

[51] R. John, J. P. Cherian, J. J. Kizhakkethottam, A survey of techniques to prevent sybil attacks, in: 2015 International Conference on Soft-Computing and Networks Security (ICSNS), 2015, pp. 1–6.

doi:10.1109/ICSNS.2015.7292385.

[52] P. Goyal, A. Goyal, Comparative study of two most popular packet sniffing tools-tcpdump and wireshark, in: 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN), 2017, pp. 77–81. doi:10.1109/CICN.2017.8319360.

[53] R. Das, G. Tuna, Packet tracing and analysis of network cameras with wireshark, in: 2017 5th International Symposium on Digital Forensic and Security (ISDFS), 2017, pp. 1–6. doi:10.1109/ISDFS.2017.7916510.

[54] G. Combs, Wireshark [https://www. wireshark. org/]. the wireshark team (2017).

[55] T. TEAM, Tcpdump/libpcap public repository (2016).

[56] M. Kershaw, Kismet wireless network sniffer, URL: kismetwireless. net.

[57] S. Bansal, N. Bansal, Scapy-a python tool for security testing, Journal of Computer Science & Systems Biology 8 (3) (2015) 140.

[58] A. S. A. M. S. Ahmed, R. Hassan, N. E. Othman, Ipv6 neighbor discovery protocol specifications, threats and countermeasures: A survey, IEEE Access 5 (2017) 18187–18210. doi:10.1109/ACCESS.2017.2737524.

[59] N. Ahmed, A. Sadiq, A. Farooq, R. Akram, Securing the neighbour discovery protocol in ipv6 state-ful address auto-configuration, in: 2017 IEEE Trustcom/BigDataSE/ICESS, 2017, pp. 96–103. doi:10.1109/Trustcom/BigDataSE/ICESS.2017.225.

[60] Sudhakar, R. K. Aggarwal, A survey on comparative analysis of tools for the detection of arp poisoning, in: 2017 2nd International Conference on Telecommunication and Networks (TEL-NET), 2017, pp. 1–6. doi:10.1109/TEL-NET.2017.8343546.

[61] J. D. Brown, T. J. Willink, Arp cache poisoning and routing loops in ad hoc networks, Mobile Networks and Applicationsdoi:10.1007/s11036-018-1039-6.
URL `https://doi.org/10.1007/s11036-018-1039-6`

[62] A. Hoehn, P. Zhang, Detection of replay attacks in cyber-physical systems, in: 2016 American Control Conference (ACC), 2016, pp. 290–295. doi:10.1109/ACC.2016.7524930.

[63] B. Chen, D. W. C. Ho, G. Hu, L. Yu, Secure fusion estimation for bandwidth constrained cyber-physical systems under replay attacks, IEEE Transactions on Cybernetics 48 (6) (2018) 1862–1876. doi:10.1109/TCYB.2017.2716115.

[64] Q. Hu, G. P. Hancke, A session hijacking attack on physical layer key generation agreement, in: 2017 IEEE International Conference on Industrial Technology (ICIT), 2017, pp. 1418–1423. doi:10.1109/ICIT.2017.7915573.

[65] Z. Lu, F. Chen, G. Cheng, S. Li, The best defense strategy against session hijacking using security game in sdn, in: 2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2017, pp. 419–426. doi:10.1109/HPCC-SmartCity-DSS.2017.55.

[66] S. Ohata, Y. Kawai, T. Matsuda, G. Hanaoka, K. Matsuura, Re-encryption verifiability: How to detect malicious activities of a proxy in proxy re-encryption, in: K. Nyberg (Ed.), Topics in Cryptology — CT-RSA 2015, Springer International Publishing, Cham, 2015, pp. 410–428.

[67] B. Libert, D. Vergnaud, Tracing malicious proxies in proxy re-encryption, in: S. D. Galbraith, K. G. Paterson (Eds.), Pairing-Based Cryptography – Pairing 2008, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 332–353.

[68] S. Singh, Y.-S. Jeong, J. H. Park, A survey on cloud computing security: Issues, threats, and solutions, Journal of Network and Computer Applications 75 (2016) 200 – 222. doi:https://doi.org/10.1016/j.jnca.2016.09.002.
URL `http://www.sciencedirect.com/science/article/pii/S1084804516301990`

[69] M. B. Mollah, M. A. K. Azad, A. Vasilakos, Security and privacy challenges in mobile cloud computing: Survey and way ahead, Journal of Network and Computer Applications 84 (2017) 38 – 54. doi:https://doi.org/10.1016/j.jnca.2017.02.001.
URL `http://www.sciencedirect.com/science/article/pii/S1084804517300632`

[70] A. Alrawais, A. Alhothaily, C. Hu, X. Cheng, Fog computing for the internet of things: Security and privacy issues, IEEE Internet Computing 21 (2) (2017) 34–42. doi:10.1109/MIC.2017.37.

[71] S. Alharbi, P. Rodriguez, R. Maharaja, P. Iyer, N. Bose, Z. Ye, Focus: A fog computing-based security system for the internet of things, in: 2018 15th IEEE Annual Consumer Communications Networking Conference (CCNC), 2018, pp. 1–5. doi:10.1109/CCNC.2018.8319238.

[72] A. J. Duncan, S. Creese, M. Goldsmith, Insider attacks in cloud computing, in: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 2012, pp. 857–862. doi:10.1109/TrustCom.2012.188.

[73] S. J. Stolfo, M. B. Salem, A. D. Keromytis, Fog computing: Mitigating insider data theft attacks in the cloud, in: 2012 IEEE Symposium on Security and Privacy Workshops, 2012, pp. 125–128. doi:10.1109/SPW.2012.19.

[74] S. William, Computer Security: Principles And Practice, Pearson Education India, 2008.

[75] J.-W. H. Bulle, L. Montoya, W. Pieters, M. Junger, P. Hartel, On the anatomy of social engineering attacksa literature-based dissection of successful attacks, Journal of Investigative Psychology and Offender Profiling 15 (1) (2018) 20–45. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/jip.1482, doi:10.1002/jip.1482.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/jip.1482

[76] J. Saleem, M. Hammoudeh, Defense methods against social engineering attacks, in: Computer and network security essentials, Springer, 2018, pp. 603–618.

[77] R. Kissel, Glossary of key information security terms, U.S. Dept. of Commerce, National Institute of Standards and Technology, 2011.

[78] C. Cowan, F. Wagle, C. Pu, S. Beattie, J. Walpole, Buffer overflows: attacks and defenses for the vulnerability of the decade, in: Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00, Vol. 2, 2000, pp. 119–129 vol.2. doi:10.1109/DISCEX.2000.821514.

[79] S. Surendran, A. Nassef, B. D. Beheshti, A survey of cryptographic algorithms for iot devices, in: 2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT), 2018, pp. 1–8. doi:10.1109/LISAT.2018.8378034.

[80] V. Drgoi, T. Richmond, D. Bucerzan, A. Legay, Survey on cryptanalysis of code-based cryptography: From theoretical to physical attacks, in: 2018 7th International Conference on Computers Communications and Control (ICCCC), 2018, pp. 215–223. doi:10.1109/ICCCC.2018.8390461.

[81] M. T. Manavi, Defense mechanisms against distributed denial of service attacks : A survey, Computers & Electrical Engineering 72 (2018) 26 – 38. doi:https://doi.org/10.1016/j.compeleceng.2018.09.001.
URL http://www.sciencedirect.com/science/article/pii/S0045790616307029

[82] P. Cichonski, T. Millar, T. Grance, K. Scarfone, Computer security incident handling guide, NIST Special Publication 800 (61) (2012) 1–147.

[83] E. Bertino, N. Islam, Botnets and internet of things security, Computer 50 (2) (2017) 76–79. doi:10.1109/MC.2017.62.
URL doi.ieeecomputersociety.org/10.1109/MC.2017.62

[84] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, Ddos in the iot: Mirai and other botnets, Computer 50 (7) (2017) 80–84. doi:10.1109/MC.2017.201.

[85] E. M. Rudd, A. Rozsa, M. Gnther, T. E. Boult, A survey of stealth malware attacks, mitigation measures, and steps toward autonomous open world solutions, IEEE Communications Surveys Tutorials 19 (2) (2017) 1145–1172. doi:10.1109/COMST.2016.2636078.

[86] Q. Hu, S. Lv, Z. Shi, L. Sun, L. Xiao, Defense against advanced persistent threats with expert system for internet of things, in: L. Ma, A. Khreishah, Y. Zhang, M. Yan (Eds.), Wireless Algorithms, Systems, and Applications, Springer International Publishing, Cham, 2017, pp. 326–337.

[87] M. Nicho, A. Oluwasegun, F. Kamoun, Identifying vulnerabilities in apt attacks: A simulated approach, in: 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2018, pp. 1–4. doi:10.1109/NTMS.2018.8328696.

[88] M. Ussath, D. Jaeger, F. Cheng, C. Meinel, Advanced persistent threats: Behind the scenes, in: 2016 Annual Conference on Information Science and Systems (CISS), 2016, pp. 181–186.

doi:10.1109/CISS.2016.7460498.

[89] N. Virvilis, D. Gritzalis, T. Apostolopoulos, Trusted computing vs. advanced persistent threats: Can a defender win this game?, in: 2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing, 2013, pp. 396–403. doi:10.1109/UIC-ATC.2013.80.

[90] A. Triantafyllou, P. Sarigiannidis, T. D. Lagkas, Network protocols, schemes, and mechanisms for internet of things (iot): Features, open challenges, and trends, Wireless Communications and Mobile Computing 2018.

[91] E. Rescorla, M. Ray, S. Dispensa, N. Oskov, Transport layer security (tls) renegotiation indication extension, Tech. rep., IETF (2010).

[92] B. Fan, Analysis on the security architecture of zigbee based on ieee 802.15.4, in: 2017 IEEE 13th International Symposium on Autonomous Decentralized System (ISADS), 2017, pp. 241–246. doi:10.1109/ISADS.2017.23.

[93] N. Vidgren, K. Haataja, J. L. Patio-Andres, J. J. Ramrez-Sanchis, P. Toivanen, Security threats in zigbee-enabled systems: Vulnerability evaluation, practical experiments, countermeasures, and lessons learned, in: 2013 46th Hawaii International Conference on System Sciences, 2013, pp. 5132–5138. doi:10.1109/HICSS.2013.475.

[94] X. Fan, F. Susan, W. R. Long, S. Li, Security analysis of zigbee, in: –, 2017, pp. 1–18.

[95] J. Wright, Killerbee: Practical zigbee exploitation framework or wireless hacking and the kinetic world (2018).

[96] B. Stelte, G. D. Rodosek, Thwarting attacks on zigbee - removal of the killerbee stinger, in: Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013), 2013, pp. 219–226. doi:10.1109/CNSM.2013.6727840.

[97] S. Marksteiner, V. J. E. Jimenez, H. Valiant, H. Zeiner, An overview of wireless iot protocol security in the smart home domain, in: 2017 Internet of Things Business Models, Users, and Networks, 2017, pp. 1–8. doi:10.1109/CTTE.2017.8260940.

[98] D. Genkin, L. Valenta, Y. Yarom, May the fourth be with you: A microarchitectural side channel attack on several real-world applications of curve25519, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, ACM, New York, NY, USA, 2017, pp. 845–858. doi:10.1145/3133956.3134029.
URL http://doi.acm.org/10.1145/3133956.3134029

[99] J. Hall, B. Ramsey, Breaking bulbs briskly by bogus broadcast, in: Presented at ShmooCon, 2016, pp. –.

[100] M. Smith, Ez-wave: A z-wave hacking tool capable of breaking bulbs, abusing z-wave devices, Network World.

[101] J. Padgette, Guide to bluetooth security, NIST Special Publication 800 (2017) 121.

[102] G. Kwon, J. Kim, J. Noh, S. Cho, Bluetooth low energy security vulnerability and improvement method, in: 2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia), 2016, pp. 1–4. doi:10.1109/ICCE-Asia.2016.7804832.

[103] S. Jasek, Gattacking bluetooth smart devices, SecuRing, Tech. Rep.

[104] A. Rose, B. Ramsey, Picking bluetooth low energy locks from a quarter mile away, DEF CON 24.

[105] S. Naoui, M. E. Elhdhili, L. A. Saidane, Enhancing the security of the iot lorawan architecture, in: 2016 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN), 2016, pp. 1–7. doi:10.1109/PEMWN.2016.7842904.

[106] E. Aras, G. S. Ramachandran, P. Lawrence, D. Hughes, Exploring the security vulnerabilities of lora, in: 2017 3rd IEEE International Conference on Cybernetics (CYBCONF), 2017, pp. 1–6. doi:10.1109/CYBConf.2017.7985777.

[107] B. Reynders, W. Meert, S. Pollin, Range and coexistence analysis of long range unlicensed communication, in: 2016 23rd International Conference on Telecommunications (ICT), 2016, pp. 1–6. doi:10.1109/ICT.2016.7500415.

[108] S. Sciancalepore, A. Capossele, G. Piro, G. Boggia, G. Bianchi, Key management protocol with

implicit certificates for iot systems, in: Proceedings of the 2015 Workshop on IoT Challenges in Mobile and Industrial Systems, IoT-Sys '15, ACM, New York, NY, USA, 2015, pp. 37–42. doi:10.1145/2753476.2753477.
URL http://doi.acm.org/10.1145/2753476.2753477

[109] P. Porambage, A. Braeken, P. Kumar, A. Gurtov, M. Ylianttila, Proxy-based end-to-end key establishment protocol for the internet of things, in: 2015 IEEE International Conference on Communication Workshop (ICCW), 2015, pp. 2677–2682. doi:10.1109/ICCW.2015.7247583.

[110] R. Riaz, K. Kim, H. F. Ahmed, Security analysis survey and framework design for ip connected lowpans, in: 2009 International Symposium on Autonomous Decentralized Systems, 2009, pp. 1–6. doi:10.1109/ISADS.2009.5207373.

[111] J. Granjal, R. Silva, E. Monteiro, J. S. Silva, F. Boavida, Why is ipsec a viable option for wireless sensor networks, in: 2008 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems, 2008, pp. 802–807. doi:10.1109/MAHSS.2008.4660130.

[112] J. Granjal, E. Monteiro, J. S. Silva, Enabling network-layer security on ipv6 wireless sensor networks, in: 2010 IEEE Global Telecommunications Conference GLOBECOM 2010, 2010, pp. 1–6. doi:10.1109/GLOCOM.2010.5684293.

[113] J. Granjal, E. Monteiro, J. S. Silva, Network-layer security for the internet of things using tinyos and blip, International Journal of Communication Systems 27 (10) (2014) 1938–1963. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/dac.2444, doi:10.1002/dac.2444.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/dac.2444

[114] S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt, U. Roedig, Securing communication in 6lowpan with compressed ipsec, in: 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011, pp. 1–8. doi:10.1109/DCOSS.2011.5982177.

[115] H. Kim, Protection against packet fragmentation attacks at 6lowpan adaptation layer, in: 2008 International Conference on Convergence and Hybrid Information Technology, 2008, pp. 796–801. doi:10.1109/ICHIT.2008.261.

[116] R. Hummen, J. Hiller, H. Wirtz, M. Henze, H. Shafagh, K. Wehrle, 6lowpan fragmentation attacks and mitigation mechanisms, in: Proceedings of the Sixth ACM Conference on Security and Privacy in Wireless and Mobile Networks, WiSec '13, ACM, New York, NY, USA, 2013, pp. 55–66. doi:10.1145/2462096.2462107.
URL http://doi.acm.org/10.1145/2462096.2462107

[117] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. P. Vasseur, R. Alexander, Rpl: Ipv6 routing protocol for low-power and lossy networks, Tech. rep., IETF (2012).

[118] L. Nastase, Security in the internet of things: A survey on application layer protocols, in: 2017 21st International Conference on Control Systems and Computer Science (CSCS), 2017, pp. 659–666. doi:10.1109/CSCS.2017.101.

[119] E. Rescorla, N. Modadugu, Datagram transport layer security version 1.2, Tech. rep., IETF (2012).

[120] R. A. Rahman, B. Shah, Security analysis of iot protocols: A focus in coap, in: 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC), 2016, pp. 1–7. doi:10.1109/ICBDSC.2016.7460363.

[121] A. Mali, A. Nimkar, Security schemes for constrained application protocol in iot: A precise survey, in: S. M. Thampi, G. Martínez Pérez, C. B. Westphall, J. Hu, C. I. Fan, F. Gómez Mármol (Eds.), Security in Computing and Communications, Springer Singapore, Singapore, 2017, pp. 134–145.

[122] D. McGrew, D. Bailey, Aes-ccm cipher suites for transport layer security (tls), Tech. rep., IETF (2012).

[123] T. Dierks, E. Rescorla, The transport layer security (tls) protocol version 1.2, Tech. rep., IETF (2008).

[124] S. Blake-Wilson, N. Bolyard, V. Gupta, C. Hawk, B. Möller, Elliptic curve cryptography (ecc) cipher suites for transport layer security (tls), Tech. rep., IETF (2006).

[125] J. Granjal, E. Monteiro, J. S. Silva, Application-layer security for the wot: Extending coap to support end-to-end message security for internet-integrated sensing applications, in: V. Tsaoussidis, A. J. Kassler, Y. Koucheryavy, A. Mellouk (Eds.), Wired/Wireless Internet Communication, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 140–153.

[126] N. Namvar, W. Saad, N. Bahadori, B. Kelley, Jamming in the internet of things: A game-theoretic perspective, in: 2016 IEEE Global Communications Conference (GLOBECOM), 2016, pp. 1–6. doi:10.1109/GLOCOM.2016.7841922.

[127] X. Tang, P. Ren, Z. Han, Jamming mitigation via hierarchical security game for iot communications, IEEE Access 6 (2018) 5766–5779. doi:10.1109/ACCESS.2018.2793280.

[128] M. Labib, S. Ha, W. Saad, J. H. Reed, A colonel blotto game for anti-jamming in the internet of things, in: 2015 IEEE Global Communications Conference (GLOBECOM), 2015, pp. 1–6. doi:10.1109/GLOCOM.2015.7417437.

[129] H. A. B. Salameh, S. Almajali, M. Ayyash, H. Elgala, Spectrum assignment in cognitive radio networks for internet-of-things delay-sensitive applications under jamming attacks, IEEE Internet of Things Journal 5 (3) (2018) 1904–1913. doi:10.1109/JIOT.2018.2817339.

[130] Y. Chen, Y. Li, D. Xu, L. Xiao, Dqn-based power control for iot transmission against jamming, in: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring), 2018, pp. 1–5. doi:10.1109/VTCSpring.2018.8417695.

[131] S. Raza, L. Wallgren, T. Voigt, Svelte: Real-time intrusion detection in the internet of things, Ad Hoc Networks 11 (8) (2013) 2661 – 2674. doi:https://doi.org/10.1016/j.adhoc.2013.04.014.
URL http://www.sciencedirect.com/science/article/pii/S1570870513001005

[132] N. Gupta, V. Naik, S. Sengupta, A firewall for internet of things, in: 2017 9th International Conference on Communication Systems and Networks (COMSNETS), 2017, pp. 411–412. doi:10.1109/COMSNETS.2017.7945418.

[133] M. Surendar, A. Umamakeswari, Indres: An intrusion detection and response system for internet of things with 6lowpan, in: 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2016, pp. 1903–1908. doi:10.1109/WiSPNET.2016.7566473.

[134] D. Midi, A. Rullo, A. Mudgerikar, E. Bertino, Kalis a system for knowledge-driven adaptable intrusion detection for the internet of things, in: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), 2017, pp. 656–666. doi:10.1109/ICDCS.2017.104.

[135] C. Cervantes, D. Poplade, M. Nogueira, A. Santos, Detection of sinkhole attacks for supporting secure routing on 6lowpan for internet of things, in: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), 2015, pp. 606–611. doi:10.1109/INM.2015.7140344.

[136] A. Wani, S. Revathi, Analyzing threats of iot networks using sdn based intrusion detection system (sdiot-ids), in: P. Bhattacharyya, H. G. Sastry, V. Marriboyina, R. Sharma (Eds.), Smart and Innovative Trends in Next Generation Computing Technologies, Springer Singapore, Singapore, 2018, pp. 536–542.

[137] N. Sultana, N. Chilamkurti, W. Peng, R. Alhadad, Survey on sdn based network intrusion detection system using machine learning approaches, Peer-to-Peer Networking and Applicationsdoi:10.1007/s12083-017-0630-0.
URL https://doi.org/10.1007/s12083-017-0630-0

[138] D. Di Francesco Maesa, P. Mori, L. Ricci, Blockchain based access control, in: L. Y. Chen, H. P. Reiser (Eds.), Distributed Applications and Interoperable Systems, Springer International Publishing, Cham, 2017, pp. 206–220.

[139] A. Ouaddah, A. A. Elkalam, A. A. Ouahman, Towards a novel privacy-preserving access control model based on blockchain technology in iot, in: Á. Rocha, M. Serrhini, C. Felgueiras (Eds.), Europe and MENA Cooperation Advances in Information and Communication Technologies, Springer International Publishing, Cham, 2017, pp. 523–533.

[140] O. J. A. Pinno, A. R. A. Gregio, L. C. E. D. Bona, Controlchain: Blockchain as a central enabler for access control authorizations in the iot, in: GLOBECOM 2017 - 2017 IEEE Global Communications Conference, 2017, pp. 1–6. doi:10.1109/GLOCOM.2017.8254521.

[141] Y. Zhang, S. Kasahara, Y. Shen, X. Jiang, J. Wan, Smart contract-based access control for the internet of things, CoRR abs/1802.04410. arXiv:1802.04410.
URL http://arxiv.org/abs/1802.04410

[142] O. Alphand, M. Amoretti, T. Claeys, S. Dall'Asta, A. Duda, G. Ferrari, F. Rousseau, B. Tourancheau,

L. Veltri, F. Zanichelli, Iotchain: A blockchain security architecture for the internet of things, in: 2018 IEEE Wireless Communications and Networking Conference (WCNC), 2018, pp. 1–6. doi:10.1109/WCNC.2018.8377385.

[143] N. Atzei, M. Bartoletti, T. Cimoli, A survey of attacks on ethereum smart contracts (sok), in: M. Maffei, M. Ryan (Eds.), Principles of Security and Trust, Springer Berlin Heidelberg, Berlin, Heidelberg, 2017, pp. 164–186.

[144] D. S. Lavrova, An approach to developing the siem system for the internet of things, Automatic Control and Computer Sciences 50 (8) (2016) 673–681. doi:10.3103/S0146411616080125.
URL https://doi.org/10.3103/S0146411616080125

[145] G. G. Granadillo, M. El-Barbori, H. Debar, New types of alert correlation for security information and event management systems, in: 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2016, pp. 1–7. doi:10.1109/NTMS.2016.7792462.

[146] C. D. Sarno, A. Garofalo, I. Matteucci, M. Vallini, A novel security information and event management system for enhancing cyber security in a hydroelectric dam, International Journal of Critical Infrastructure Protection 13 (2016) 39 – 51. doi:https://doi.org/10.1016/j.ijcip.2016.03.002.
URL http://www.sciencedirect.com/science/article/pii/S187454821630035X

[147] P. Sarigiannidis, E. Karapistoli, A. A. Economides, Modeling the internet of things under attack: A g-network approach, IEEE Internet of Things Journal 4 (6) (2017) 1964–1977. doi:10.1109/JIOT.2017.2719623.

[148] A. OSSIM, The open source siem, URL: https://www. alienvault. com/products/ossim (accessed 25.06. 2017).

Table 10: Security Attributes of the IoT Communication Protocols.

Protocol	Confidentiality	Integrity	Availability	Authenticity	Protection against Replay Attacks	Access Control	Vulnerabilities or Limitations
IEEE 802.15.4	✓	✓	X	✓	✓	✓	-It cannot protect ACK messages -It cannot implement all keying models -Inefficient access control -Vulnerable against jamming attacks
ZigBee	✓	✓	X	✓	✓	✓	-Vulnerable against replay attacks -Vulnerable against jamming attacks -Inefficient access control -Key management issues
Z-Wave	✓	✓	X	✓	✓	✓	-Few vulnerabilities in specific applications -Vulnerable against jamming attacks
BLE	✓	✓	X	✓	X	X	-Vulnerable against jamming attacks -Vulnerable against replay attacks -Key management issues
LoRaWAN	✓	X	X	X	X	X	-Vulnerable against jamming attacks -Vulnerable against replay attacks -Key management issues
6LoWPAN	X	X	X	X	X	X	It does not provide any security measure
RPL	✓	✓	X	X	✓	X	-Vulnerable against network attacks
TLS	✓	✓	X	✓	✓	✓	Heavy for IoT applications

DTLS	✓	✓	X	✓	✓	✓	Various limitations for IoT applications
CoAP	via DTLS	via DTLS	X	via DTLS	via DTLS	via DTLS	It depends only on DTLS