


Article

A Comprehensive Study of Security and Privacy Guidelines, Threats, and Countermeasures: An IoT Perspective

Hezam Akram Abdul-Ghani *  and Dimitri KonstantasGeneva School of Economics and Management, Geneva University, 1211 Geneva, Switzerland;
Dimitri.Konstantas@unige.ch

* Correspondence: mohammed.akram@unige.ch

Received: 14 February 2019; Accepted: 9 April 2019; Published: 22 April 2019



Abstract: As Internet of Things (IoT) involvement increases in our daily lives, several security and privacy concerns like linkability, unauthorized conversations, and side-channel attacks are raised. If they are left untouched, such issues may threaten the existence of IoT. They derive from two main reasons. One is that IoT objects are equipped with limited capabilities in terms of computation power, memory, and bandwidth which hamper the direct implementation of traditional Internet security techniques. The other reason is the absence of widely-accepted IoT security and privacy guidelines and their appropriate implementation techniques. Such guidelines and techniques would greatly assist IoT stakeholders like developers and manufacturers, paving the road for building secure IoT systems from the start and, thus, reinforcing IoT security and privacy by design. In order to contribute to such objective, we first briefly discuss the primary IoT security goals and recognize IoT stakeholders. Second, we propose a comprehensive list of IoT security and privacy guidelines for the edge nodes and communication levels of IoT reference architecture. Furthermore, we point out the IoT stakeholders such as customers and manufacturers who will benefit most from these guidelines. Moreover, we identify a set of implementation techniques by which such guidelines can be accomplished, and possible attacks against previously-mentioned levels can be alleviated. Third, we discuss the challenges of IoT security and privacy guidelines, and we briefly discuss digital rights management in IoT. Finally, through this survey, we suggest several open issues that require further investigation in the future. To the best of the authors' knowledge, this work is the first survey that covers the above-mentioned objectives.

Keywords: Internet of Things; security guidelines; privacy guidelines; countermeasures; security goals; attacks

1. Introduction

Despite being widely-used, the term Internet of Things is still fuzzy, since it is composed of so many technologies like Wireless Sensor Networks (WSN), Radio-Frequency Identification (RFID), and Machine-to-Machine communications (M2M). IoT involvement in our daily lives is witnessing drastic growth and development which can be noticed in IoT applications such as smart cities, smart cars, smart buildings, and home automation. It is because of this innovation named IoT that infinite services and solutions are getting bigger and bigger. However, providing such solutions in a reliable and secure way is not straightforward, as IoT systems inherit most of the problems of the current Internet and, most probably, intensify them due to direct association with physical objects [1].

Being tightly integrated with human beings and their environments, IoT can be misused and a single weakness may result in harmful impacts such as physical damage, financial losses, privacy

infringement, and organized crime. If left unchecked, these effects may hinder IoT in reaching its full potential and growth. Most security threats and widespread privacy issues derive from two main reasons. One is that IoT objects are equipped with limited resources in terms of energy consumption, memory capacity and computational power [2]. Such limited capabilities may impede the direct implantation of conventional Internet techniques like the Advanced Encryption Standard (AES) into the IoT [3,4]. End-to-end secure communications in rich-resource objects such as laptops, tablets and phones, for instance, can be accomplished by either transport layer through Transport layer security (TLS) or network layer through Internet Protocol Security (IPsec). Nevertheless, these techniques cannot directly be implemented in limited-resource objects, and their absence may lead to various security and privacy attacks like eavesdropping, network side-channel attacks, and tracking. Hence, the adaptation of previously-mentioned techniques (TLS and IPsec) into IoT paradigm is paramount so as to cope with those attacks and, at the same time, meet IoT requirements. Regarding this point, several research proposals, discussed in Section 4, have been introduced in relation to this topic [5–9].

The other reason, which is the most important one, is the inadequacy of well-investigated security and privacy guidelines and their proper implementation techniques. Such guidelines and techniques would greatly help IoT stakeholders (e.g., developers and manufacturers), paving the road for building secure IoT systems from the start and, thus, enhancing IoT security and privacy by design [10]. Overlooking these guidelines and their proper countermeasures in IoT system development life cycle (SDLC) may result in several attacks and threats. For example, one guideline may suggest using a permanent hardware secure boot process in order to prevent an IoT system from running a malicious code. However, if ignored by manufacturers, an adversary could easily compromise their objects by replacing the existing executable files with malicious ones [11]. In spite of the importance of IoT security and privacy guidelines and their implementation techniques in protecting IoT systems, a few research efforts, most of them white papers, have been conducted in the state-of-the-art, described as follows:

In [12], the Broadband Internet Technical Advisory Group (BITAG) proposes a high-level set of IoT security and privacy guidelines for communication, data, service, and object. However, BITAG does not provide a comprehensive list of guidelines, nor does it describe the required countermeasures to implement its guidelines. Furthermore, attacks and threats against IoT remain untouched.

In [13], the authors propose a set of privacy guidelines for IoT applications and middle-ware platforms used to evaluate two open IoT middle-ware platforms: OpenIoT and Eclipse SmartHome. Nevertheless, they uncover appropriate countermeasures to apply their guidelines and feasible attacks against IoT. The United States Department of Homeland Security (USDHS) in [14] suggests a set of security best practices as well as a list of principles to raise awareness among IoT stakeholders to improve IoT security. Such practices and principles are not comprehensive. However, USDHS does not investigate attacks and threats against IoT, nor does it identify suitable security solutions to execute its best practices. The IoT Security Foundation (IoTSF) in [15] has proposed a comprehensive list of security and privacy guidelines for IoT applications, operating systems, object hardware, wireless and wired interfaces, cloud, networks, and mobile applications. Moreover, the IoTSF has stated those who may benefit most of its guidelines. However, IoTSF does not distinguish proper countermeasures to achieve its guidelines, or discuss potential attacks and threats against IoT. The Open Web Application Security Project (OWASP) in [16] has suggested a set of IoT security guidance for different IoT aspects including, but not limited to, network services, authentication/authorization, cloud interface, physical object. Furthermore, the OWASP has identified those IoT stakeholders like manufactures, customers, and developers who may use its guidance to enhance IoT security. The OWASP, however, uncovers both countermeasures, and attacks and threats against IoT. The European Union Agency for Network and Information Security (ENISA) in [17] has distinguished a list of baseline guidelines used to alleviate threats and possible attacks against IoT. The main purpose of its work is to offer a clear understanding of IoT security requirements and identify in an abstract level some of IoT attacks and threats against

its proposed IoT asset taxonomy. Th ENISA, however, does not state those who may benefit from its guidelines, or recognize appropriate techniques to carry out them.

By observing Table 1 that summarizes the above-mentioned research proposals conducted in this area, it is not hard to see the many shortcomings while going through them. Thus, our research is conducted and designed to overcome those limitations that can be classified as follows: (i) The lack of a thorough list of security and privacy guidelines for IoT ecosystem, followed by to whom these guidelines are intended for. (ii) The absence of suitable countermeasures to carry out such guidelines. (iii) The need for IoT attack investigations.

Table 1. Comparison of research efforts presented in the literature.

Addressed Features		State-of-the Art Work						This Work
		[13]	[12]	[16]	[17]	[14]	[15]	
IoT Asset Guidelines	Computing nodes	✗	✓	✓	✓	✓	✓	✓
	RFID tags	✗	✗	✗	✗	✗	✓	✓
	Protocols	✓	✓	✗	✓	✗	✓	✓
Types of Guidelines	Privacy	✓	✓	✓	✓	✗	✓	✓
	Security	✗	✓	✓	✓	✓	✓	✓
Guidelines Intended for	Manufacturer	✗	✗	✓	✗	✓	✗	✓
	Developer	✓	✓	✓	✓	✓	✓	✓
	Customer	✗	✗	✓	✗	✗	✗	✓
	Provider	✗	✗	✗	✗	✓	✗	✓
Threats Mitigated by Guidelines		✗	✗	✗	✗	✗	✗	✓
Technique to implement Guid		✗	✗	✗	✗	✗	✗	✓

To this point, the first step towards addressing such limitations is to identify precisely IoT asset which is too complex due to its combination of a large number of technologies. Nevertheless, several IoT reference models (RM) such as a three-level model [18], a five-level model [19], a four-level model [20], and a seven-level model [21] have been proposed in the state-of-the art.

In order to propose a thorough set of security and privacy guidelines along with their implementation techniques and also identify all possible attacks against edge nodes and communication levels, we use CISCO's RM proposed in 2014 as a thorough extension of the five-level and the three-layer models in this work, depicted in Figure 1. This is because such RM simplifies the complexity of IoT ecosystem by breaking it down into different levels, clarifies IoT by providing additional information to recognize each level, organizes IoT by making it real and approachable, and standardizes IoT by offering a first step in allowing different manufacturers to develop IoT products which can communicate with each other [21]. Next, we briefly discuss each level of this model.

- **Level 1-Edge nodes:** This level is composed of computing nodes such as sensors, micro-controllers, RFID readers, and several types of RFID tags. Several security goals like integrity, confidentiality, and privacy should be taken into consideration from this level upwards.
- **Level 2-Communication:** This level consists of all enabler technologies (e.g., connectivity and communication protocols) which allow transportation of commands and data between objects in the first level and objects located at the third level.
- **Level 3-Edge computing:** The main objective of this level is to perform simple data processing which in turn decreases the computation load in the higher level and offers a quick response. It is wise for real-time applications to process data closer to the edge of the network, rather than to process data in the cloud. Many factors (e.g., service providers and computing nodes) can be used to define the amount of data processing at this level.

- Level 4-Data accumulation: As most of IoT applications may not require immediate data processing, this level converts data in motion to data at rest. It provides several functions, the most popular of which are changing packets to database tables, deciding if data is of importance to higher levels, and minimizing data via filtering process.
- Level 5- Data abstraction: This level is used to store data for further processing. In general, this level provides several functions such as normalization/denormalization, indexing, and access control to different data centers.
- Level 6-Applications: The list of IoT applications operated at this level is almost endless in both sectors (industries and markets). Information interpretation can be provided as a result of cooperation between different applications, which in turn depends on data either at rest or in motion at this level.
- Level 7- Data centers (DC) and Users: In this level, only authorized users should be allowed to communicate with IoT applications to make use of their data. Such data may be stored remotely in DCs for further processing.



Figure 1. CISCO's RM [21].

This survey analyzes in depth the first two levels of CISCO'S RM (edge nodes and communication) to provide a big picture of the current state of the art on IoT attacks, guidelines, and countermeasures. More specifically, it:

- introduces a comprehensive list of security and privacy guidelines for edge nodes and communication which can be used to enhance IoT security and privacy by design;
- reviews a set of countermeasures in which such guidelines can be implemented and also states those IoT stakeholders who will benefit from these guidelines;
- investigates in depth possible attacks and threats against edge nodes and communication;
- suggests some recommendations and open challenges for future work with the aim of directing researchers to areas that require further investigation.

As a summary, the major contribution we intend to produce lies in reinforcing IoT security and privacy by design by given IoT stakeholders and researchers an opportunity to integrate such guidelines along with their proper implementation techniques from the early stages of their systems.

The rest of the work is organized as follows. In Section 2, we discuss IoT security goals and identify IoT stakeholders. In Section 3, we introduce a complete set of security and privacy guidelines for edge nodes and describe possible attacks at this level. Moreover, we recognize proper countermeasures to apply the suggested guidelines at this level. In Section 4, we provide a full set of security and privacy guidelines for communication and also describe possible attacks at this level. Furthermore, we identify proper countermeasures to implement the suggested guidelines at this level. Digital rights managements in IoT and challenges of IoT security guidelines are discussed in Sections 5 and 6, respectively. Finally, we also suggest some open issues that need to be addressed in In Section 7.

2. IoT Security Goals and Stakeholders

In this section, we first discuss IoT security goals and, then, identify IoT stakeholders.

A. IoT security goals: In the literature, traditional security goals are divided into three major groups: (i) confidentiality, (ii) integrity, and (iii) availability, known as a CIA-triad. Confidentiality ensures that sensitive information can only be accessed by authorized objects or users. With the advent of IoT, it is essential to guarantee the confidentiality of IoT objects, as they may deal with sensitive information like credit cards and medical records. For example, the authors in [22] illustrate the impacts of an authorized access to medical objects which may expose personal information or result in life-threatening cases. In an IoT context, integrity is also essential for providing reliable solutions in which only valid commands and data are received. Integrity compromise can result in harmful consequences. For instance, the authors in [23] describe successful attacks against an Insulin Pump which can reveal patients' privacy. IoT availability is crucial to assure that IoT services are available and cannot be interrupted. Despite the popularity of the CIA-triad, the authors in [24] prove its insufficiency of addressing novel threats, emerging in a collaborative environment. To cope with this issue, they offer a thorough set of security goals called information, assurance, and security (IAS) octave, referred to as the IAS-octave, by examining a huge number of information in literature in terms of security. Table 2 highlights the security goals suggested by the IAS-octave, along with their definitions and abbreviations.

Table 2. Internet of Things (IoT) Security goals [20].

Security Requirements	Definition	Abbreviations
Confidentiality	The process in which only authorized objects or users can get access to the data	C
Integrity	The process in which data completeness, and accuracy is preserved	I
Non-repudiation	The process in which an IoT system can validate the incident or non-incident of an event	NR
Availability	An ability of an IoT system to make sure its services are accessible, when demanded by authorized objects or users	A
Privacy	The process in which an IoT system follows privacy rules or policies and allowing users to control their sensitive data	P
Audibility	Ensuring the ability of an IoT system to perform firm monitoring on its actions	AU
Accountability	The process in which an IoT system holds users taking charge of their actions	AC
Trustworthiness	Ensuring the ability of an IoT system to prove identity and confirm trust in third party	TW

B. IoT stakeholders: We classify IoT stakeholders into four categories based on their roles in this complex ecosystem, varying from objects located in the physical world and their data in the cloud. Table 3 outlines IoT stakeholders and gives their roles and abbreviations.

Table 3. IoT stakeholders.

IoT Stakeholders	Roles	Abbreviations
Manufacturer	Building IoT products, and IoT hardware	M
Developer	Developing IoT solutions either from scratch or from open-source components	D
Consumer	Using IoT objects in different aspects of their daily lives	C
Provider	Providing IoT services to customers	P

3. Level 1: Edge Nodes

In this section, we first introduce a comprehensive IoT security and privacy guidelines for the first level of CISCO's RM (edge nodes) composed of computing nodes (RFID readers and sensor nodes) and RFID tags. Then, we investigate potential attacks and threats against this level. Finally, we distinguish proper countermeasures found in literature to implement our proposed guidelines and also address attacks at this level.

3.1. Security and Privacy Guidelines for Computing Nodes

Figure 2 summarizes the relationship between proposed guidelines for computing nodes, their suitable countermeasures, and their possible attacks. Next, we briefly discuss required security and privacy guidelines suggested for computing nodes.

- **Prevent node replication:** In [25], the authors have stated that node replication attack can be launched by replicating object's identification number in any IoT networks leading to a huge drop in the network performance. Therefore, this guideline suggests that each IoT object should have right means to protect its identification number.
- **Secure boot process:** This guideline suggests that each IoT object should be equipped with a fixed hardware secure boot process designed to prevent an IoT system from running a malicious code. Before an IoT object starting, a malicious code could be loaded and executed in the boot process, leading to install a "bootkit" used to maintain the malicious code and also control the object to mask its existence [15]. For example, in the absence of the immutable hardware secure boot an attacker could compromise an IoT object if he could find a method to replace its firmware with malicious one.
- **Secure debug ports such as JTAG:** Test Access Port and Boundary-Scan Architecture (JTAG port), which is an IEEE standard, has been developed as a useful interface for embedded systems to achieve many purposes like test, debug and development [26]. This guideline suggests preventing a simple access to IoT objects from external world and also monitor their scan chains using a secure JTAG. The JTAG port, however, can be easily compromised by attackers [27].

For interested readers, the recent survey published in this regard can be found in [28].

- **Create unique security parameters:** This guideline suggests that security parameters like private cryptographic keys, and passwords for each IoT object should be unique. The key benefit from this guideline stems from disclosure of security parameters on one IoT object cannot be used to compromise other objects [29].
- **Change default password:** A default password can be considered one of the simplest security methods and, at the same time, can be a source of major security issues (e.g., botnet attacks) [30]. Thus, this guideline suggests that each IoT object must be equipped with a technique with which its default password must be changed on first use [10].

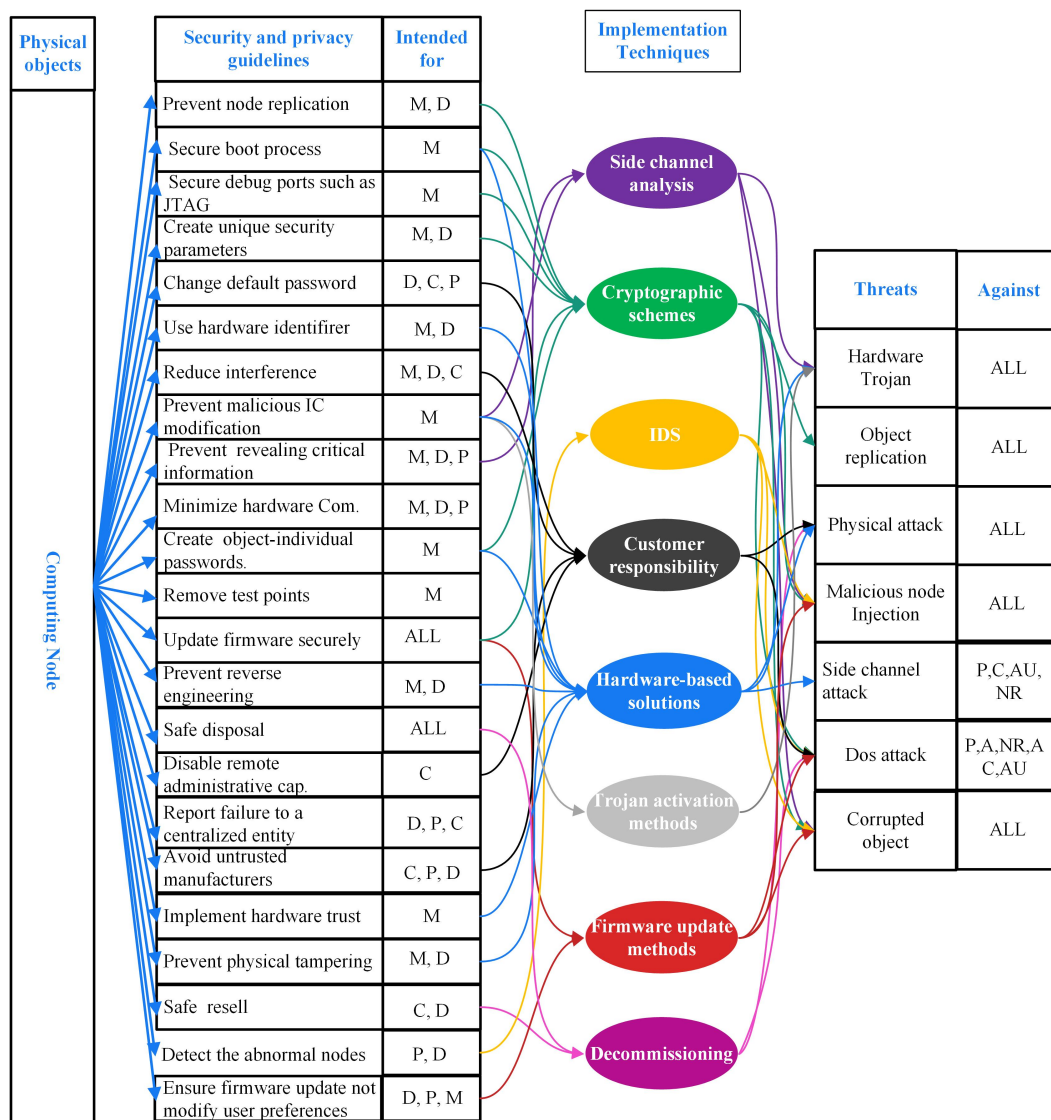


Figure 2. An overview of guidelines, stakeholders, attacks and countermeasures for computing nodes.

- **Use hardware identifier and authentication:** This guideline suggests that each IoT object should be integrated with a unique unforgettable identity not separable from its hardware [10]. An IoT object can store its ID in the form of credential developed during manufacturing inside its a secure element [31]. This ID uniquely identifies the object as well as its proof of origin [32].
- **Reduce interference:** This guideline suggests minimizing the interference between IoT objects (computing nodes and RFID tags) and ensure reliable performance, as most IoT objects are armed with wireless technologies which may cause a node jamming [33]. For instance, an attacker could carry an object which actively propagates radio signals in order to block or interrupt the functions of any neighboring objects [34].
- **Prevent unwanted IC modifications:** This guideline suggests that each IoT object should be able to detect any malicious modifications on its integrated circuit (IC), preventing an attacker from exploiting its functionality to gain access to its data. If implemented, such a guideline would prevent hardware Trojan attacks [35].
- **Prevent revealing critical information:** This guideline suggests that each IoT object should be shielded with specific techniques such as a side-channel analysis to prevent unauthorized attempts to reveal its critical information. Different patterns like power analysis can be used by an attacker to expose sensitive information of an object, even when its messages are encrypted [36].

- **Minimize hardware components:** To minimize IoT attack surface and avoid unwanted threats, IoT objects should be functioning on the principle of least privilege [37]. Therefore, not only unused ports (e.g., USB ports), but also unnecessary hardware components must be closed for each IoT object [15].
- **Create an object-individual password:** This guideline suggests that each IoT object should have a unique password, and it must not be re-settable to any global factory default value. If it is sold with the default password and user-name (e.g., admin) and if these are not changed by customers, an IoT object will face many security and privacy issues [38]. For instance, allowing customers to access IoT objects, manufacturers integrate their objects with a telnet and web interface enabled by default user-names and passwords. Unfortunately, such objects with default credentials are vulnerable to several IoT malwares (e.g., Carna, LightAidra/Aidra) [30].
- **Remove test points:** After manufacturing an IoT object, its test points must be secured so that they cannot be used by any attacker to compromise its confidentiality and integrity [15].
- **Update firmware securely:** In the context of IoT, a firmware update takes place either remotely or directly. A remote firmware update depends heavily on a base station with which a new version of firmware is broadcasted to all nodes willing to update their firmware. In this case, this guideline suggests that a new firmware image should be encrypted, and its integrity should be checked. On the other hand, a direct firmware update (e.g., using a USB cable) depends on an end user. In this case, this guideline suggests that the end user must be authenticated [35].
- **Prevent reverse engineering:** As most IoT objects may be placed in unattended environments, such objects are susceptible to physical attacks (e.g., a reverse engineering). An attacker, for instance, could get access to an object and, then, he/she could take it apart to discover its main components and security parameters. This guideline, therefore, suggests that each IoT object should be armed with a tamper-proofing technique to resist reverse engineering attacks [15].
- **Safe disposal:** IoT objects, through their life cycles, may be deployed in different environments to perform specific tasks. They also may change their environments several times, reaching a point in which such objects should be disposed or removed from services without revealing their sensitive information [39]. If not destroyed properly, objects sensitive data along with security parameters could be reversed back by an attacker [40]. This guideline, therefore, suggests that manufacturers should provide a clear end-of-life strategy along with a formal plan in which obsolete objects can be disposed by customers without revealing their sensitive data [14].
- **Disable remote administrator capabilities:** Many IoT objects should have remote administrative capabilities due to the nature of IoT technology in which objects may be placed in remote environments. Despite the benefit of using remote administrative capabilities to monitor and control IoT objects, they could open a door to many attacks (e.g., Man in middle attack). This guideline, therefore, suggests that remote administrative capabilities should not be enabled by default for all IoT objects.
- **Report failures to a centralized entity:** To build global efforts towards mitigating IoT threats and reaching the full potential of IoT, each IoT object should be equipped with an error reporting technique to report each failure of objects to a manufacture or a centralized entity [41].
- **Avoid untrusted manufactures:** The increasing demand of IoT solutions has led to the existence of so many manufactures, among which are untrusted ones. This guideline, thus, suggests that IoT customers and developers should avoid buying IoT components or products from untrusted manufacturers [15].
- **Implement hardware trust:** Trust data in IoT systems is an indispensable requirement, since IoT systems are designed to interact with each other to achieve certain tasks. If data of a single sensor has been maliciously modified, the whole IoT system may be considered insecure [31]. For instance, a humidity sensor may be modified to always send a specific value irrespective of the actual one. This guideline, therefore, suggests the use of a hardware trust in each object like a Physically Unclonable Function (PUF).

- **Prevent physical tampering:** In some cases, IoT objects may be deployed and operated in either remote or hostile environments in which a direct access to such objects may be possible, making them susceptible to hardware/firmware attacks. This guideline, therefore, suggests that each IoT object should be equipped with a suitable tamper resistant measure [42].
- **Safe resell:** There are some situations in which IoT objects may be resold to other customers. In this context, such objects must not reveal the sensitive data of the previous customers. This guideline suggests that each IoT object should be armed with a suitable wiping-out mechanism [12].
- **Detect the abnormal nodes and sensors:** The IoT ecosystem is identified as a network of networks. Some of these networks may be deployed in unattached environments, making them an easy target for some attackers by modifying the existing nodes to behave maliciously. It is, therefore, essential that each IoT network has capabilities to detect abnormal nodes or sensors which may cause harmful consequences to the whole network [43].
- **Firmware update not modify user preferences:** As new security measures have been introduced in IoT, hackers will find some vulnerabilities to breach such security solutions. Therefore, objects' firmware need to be updated continuously without modifying users' preferences [43]. For instance, if a user enables a secure JTAG, any firmware updates which may be accomplished automatically by an object cannot disable the secure JTAG.

3.2. Security and Privacy Guidelines for RFID Tags

Figure 3 summarizes the relationship between proposed guidelines for RFID tags, their suitable countermeasures, and their possible attacks. Next, we briefly describe required security and privacy guidelines suggested for RFID tags, some of which (common ones) have been already discussed in Section 3.1.

- **Provide distance-based information:** This guideline suggests that each RFID tag should provide its information to a reader if and only if it lies within its predefined distance. For instance, if a tag is scanned at 10 m, the tag may publish only public information, but if the tag is scanned with 1 m distance, it provides its unique identifier.
- **Secure kill command for tags:** Being designed with a kill command, a unique PIN (e.g., a 23-bit password), during manufacturing process, RFID tags can be killed by their reader if they received the correct PINs. This guideline suggests that the kill command in each RFID tag should be secured and cannot be killed by unauthorized readers [44]. For example, the isolation of tags as well as blocking can be considered as a direct way to protect a secure kill command, as attackers cannot reach such tags.
- **Secure a manufacture and a product code on EPC tag:** There are some kinds of tags which may have on-board sensitive or valuable information about objects and humans attached to. These types of tags are called Electronic Product Code (EPC) tags, consisting of two components: a manufacturer code and a product code. As a consequence, people having EPC tags are susceptible to inventorying attacks [45]. It is, thus, essential to secure such types of tags.
- **Check All readers' request to tags:** The authors in [46], have stressed the importance of examining or checking all readers' request to RFID tags to prevent unwanted scanning.
- **Change an anonymous ID frequently** Even though the authors in [47] propose a novel approach based on a look-up table to prevent attackers from discovering real IDs tags after converting them to anonymous ones, attackers can still track RFID systems as long as anonymous IDs are not changed over time.
- **Prevent tag counterfeiting:** According to [48], the only one condition in which an attacker could counterfeit a tag in a RFID system is to modify the tag's identity by methods of tag manipulation. So, this guideline suggests that each RFID tag should be armed with a lightweight anti-counterfeit technique to protect its identity.

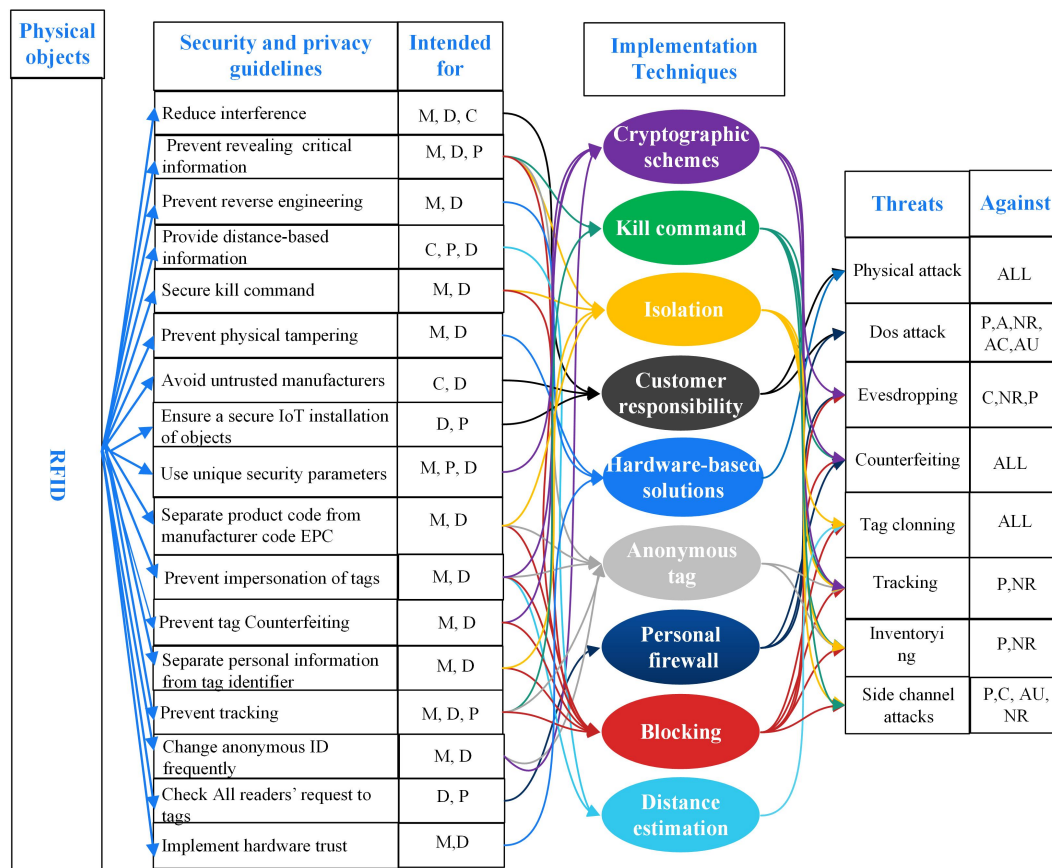


Figure 3. An overview of guidelines, stakeholders, attacks and countermeasures for Radio-Frequency Identification (RFID) tags.

- **Separate personal information from tag identifier:** In [49], the authors stated that threats and attacks against RFID systems could be grown exponentially if they combined tags' identifiers with personal information. This guideline, therefore, suggests that personal information (e.g., credit card and personal profile) should be separated from tags identifiers.
- **Prevent tracking:** As most RFID tags have unique identifiers attached to physical objects or individuals, an attacker could track their information. This guideline, thus, suggests that tags' identifiers should not be read by unauthorized readers [49].

3.3. Possible Attacks and Threats against Edge Node Level

In what follows, we discuss several attacks against computing nodes and RFID tags.

- **Hardware Trojan:** One of the major security issues for ICs is hardware Trojans. They maliciously modify ICs to allow attackers to exploit their functionalities and gain access to software operating on them [50]. To insert a Trojan into an IC, an attacker modifies maliciously the IC during its design and defines a trigger technique to initiate malicious functions of the Trojan. Trojans based on their activation methods can be classified into two categories: (i) an internally-triggered Trojan activated after a certain event is met [51] and (ii) an externally-triggered Trojan activated by a sensor or an antenna.
- **Node replication:** The main goal of such an attack is to add maliciously an object by duplicating one object's identification number to a current set of objects. A remarkable drop in network performance can happen as a consequence of this attack. Furthermore, upon arrival of packets at a replica, it may not only corrupt the packets, but also misdirect them, causing serious damage to IoT systems by allowing an attacker to gain access to security parameters (e.g., shared keys). It is also capable of revoking authorized nodes, since it can carry out an object-revocation protocol [25].

- **Denial of Service (DoS) attacks:** DoS attacks in computing nodes can be classified into three categories: sleep deprivation, outage, and battery draining attacks. In what follows, we briefly describe each one.

1. Sleep deprivation: It is a specific kind of Dos attack in which a battery-operated node may receive a huge number of requests, which look like legitimate ones, sent by an attacker. The detection of such attack, therefore, is extremely hard. In [52], Stajano coined the term “sleep deprivation”. The consequences of sleep deprivation attacks on resource-constrained objects can be found in [53].

2. Battery draining: IoT objects are equipped with small batteries because of size limitations. This is why battery-draining attacks are extremely powerful, leading to harmful impacts, such as a power outage. For instance, having drained a battery of a smoke detector by any techniques (e.g., sending tons of requests), an attacker would be able to deactivate a fire detection system [54].

3. Outage attacks: This type of attacks takes place when an IoT object stops carrying out its essential functions. This might have happened due to undesired error in the manufacturing phase, sleep deprivation, and code injection. A common example of an outage attack is what happened to Iran’s nuclear control system, when it is injected by Stuxnet [55]. This malicious worm modifies nuclear control system in such a way that it cannot detect unusual behavior, preventing it from shutting down even in case of danger.

- **Physical attack:** As the nature of IoT may require the deployment of some objects in hostile environments, such objects are vulnerable to physical access which may lead to hardware/firmware attacks. With physical access to an object, an attacker can derive precious cryptographic information, alter operating system, and vandalize circuit, all of which may result in long-term destruction. A recent example of such attack takes place in a Nest thermostat, when its firmware was replaced with a malicious one enabling the attacker to dominate the thermostat [56].

It is worth noting that physical attacks could also have happened on RFID tags if an attacker has physical access to them. In this case, the attacker could take such tags to his/her laboratory to alter and manipulate them. Many attacks against RFID tags have been identified in the literature. The most popular of them are circuit modification, clock glitching, and material removal [48].

- **Malicious node:** Obtaining unauthorized access to an IoT network and other objects, an attacker can inject a malicious object, resulting in controlling the network by the attacker. It is also possible that the attacker can inject false data into the network which may impede the delivery of valid messages.
- **Side channel attack:** While IoT objects perform their normal operations, there is a possibility that such objects might disclose critical information. This takes place even when they are not using wireless protocols to transfer data. For instance, an electromagnetic (EM) wave released by an object can provide precious information about the state of the object. This concept is called a non-network side-channel attack started to develop by TEMPEST documents [57] in 2007 as well as a set of current publications [58,59]. For instance in [59], the authors illustrate how EM emitted from a medical object can release valuable information about not only the object but also the patient.

In RFID technology, a side-channel attack also takes place even when messages are encrypted. In this case, an attacker can use a ready-developed tool to intercept communication between tags and a reader to elicit information from different patterns. For example, having read the tags at the entrance of a home, an attacker can estimate the number of people in the home [59].

- **Eavesdropping:** Even though an eavesdropping attack is commonly associated with communication protocols, it is possible to take place at this level, specifically for RFID tags. The main objective of such attack is to intercept, read, and modify messages for further investigation. The threats posed by eavesdropping in RFID tags have been investigated in many recent reports including, but not limited to, USDHS [14], the Federal Trade Commission [60], and the Cloud Security Alliance [10]. Besides these reports, several surveys published can be found in [44,48]. In [61], the authors have discussed some practical attack cases along with their experimental settings.
- **Tag Counterfeiting:** In this type of attacks, an attacker could modify the identity of an object using techniques of tag manipulation. Unlike a cloning attack, which needs more information to launch it, counterfeiting attack requires less information to initiate. In such attack, a tag is partially modified [20].
- **Tag cloning:** This type of attacks could be extremely valuable to hackers and, at the same time, could be too dangerous for company's reputation. By cloning tags, an attacker can gain access to sensitive data and closed areas [62].
- **Tag Tracking:** Tracking tags is one of the most common threats against RFID tags. This is because all tags have unique identifiers. A malicious reader can simply read a tag attached to an individual or an object, leading to strong tracking information [63]. Using a large number of readers by an attacker to read fixed tags' identifiers is the easiest form of such attack. Combining tag identifier with personal information will amplify the threats of this attack.
- **Tag Inventorying:** Different types of tags containing sensitive data may attach to many objects. To this point, an EPC is composed of two fields: a product code and a manufacturer code. As a consequence, people having EPC tags are vulnerable to inventorying [48]. For instance, by knowing what type of a medical object is attached to a patient (e.g., a insulin pump), an attacker can identify his/her sickness.

3.4. Implementation Techniques Suitable for Edge Nodes

Next, we describe countermeasures for implementing our proposed security and privacy guidelines and mitigating possible attacks against the edge nodes and RFID tags.

- **Side channel analysis:** It offers a powerful method to detect not only hardware Trojans but also malicious firmware on IoT objects.

1. Hardware Trojan Detection: To detect hardware Trojan, several side-channel signals, such as power [64,65], timing [66,67], and spatial temperature [68], have been suggested. The existence of a Trojan in an IoT object or a circuit has some impacts on its components, the most common of which are on power and gates. Moreover, it could even modify heat distribution on the IC. By comparing physical features as well as heat distribution map for a suspicious IC to a Trojan-free IC, hardware Trojan can be detected. The detection of Trojans in Timing-based method depends heavily on checking the IC using delay tests, while the detection of Trojans in power-based method depends entirely on continuous monitoring of the IC. Spatial temperature-based uses infrared imaging methods to detect the Trojans by providing thermal maps of the IC.

In addition to above-mentioned methods, Jaya et al. [36] propose a lightweight dynamic permutation technique to protect integrated circuits from both Trojan and side-channel attacks by dynamically changing the real order of data coming from sensors.

For interested readers, the recent surveys of hardware Trojan taxonomy and detection can be found in [69,70].

2. Malicious Firmware Detection: The feasibility of side-channel analysis in detecting malicious firmware has been illustrated by many previous research works [71–73]. Like hardware Trojan detection, malware detection techniques can analyze side-channel signals to discover abnormal

behaviors of IoT objects. For instance, such techniques can detect a malicious installation of firmware on an object if there is a dramatic increase on its power consumption.

- **Cryptographic schemes:** Three different types of cryptographic techniques, namely encryption, hash-based functions, and lightweight protocols, are widely used in the literature to implement some of our proposed security and privacy guidelines and mitigate possible threats at this layer. It is worth mentioning that in this section, we describe only cryptographic techniques suitable for computing nodes and RFID (others will be discussed at communication level)

1. Encryption: Both symmetric and asymmetric encryption can be used to achieve several security and privacy guidelines suggested in this paper. Many solutions for securing an object's boot-loader based on symmetric-key techniques have been proposed in the literature [11]. On the other hand, a few solutions for securing an object's boot-loader based on asymmetric-key techniques have been proposed [27]. Several solutions for securing an object's JTAG based on symmetric-key techniques have been proposed in the literature [74–76]. A recently published survey in this regard can be found in [28].

In RFID tags, a direct implementation of full encryption techniques is not feasible because of the necessity for tags to be low-cost (e.g., 10 cents), which limits their computational power and memory. It is worth noting that a standard implementation of AES requires 20–30k gates, while RFID tags can support 5–10k gates [77]. However, Jung et al. in [78] have proposed a new implementation of AES which needs only 3595 gates. Recently-proposed technique for RFID encryption can be found in [79]. That said, there is a lack of a fully-designed version of AES in RFID tags.

2. Hash-based techniques: Being widely used for investigating security issues of RFID systems, several solutions have been proposed in literature [80–82]. In [49], the authors have suggested several security techniques for RFID, one of which is based on hash function. Such a technique introduces two states of each tag: (i) locked state through which a tag answers all queries with its hashed value and (ii) unlocked state through which the tag performs its usual operation.

3. Lightweight protocols: Developing RFID tags at low cost is an indispensable requirement in RFID technology, which hinders the implementation of traditional cryptographic techniques. However, many lightweight cryptographic protocols have been proposed [83,84]. For instance, the authors in [83] suggest a lightweight mutual authentication protocol for RFID tags, the implementation of which requires only 300 gates. More importantly, the authors claim that this protocol offers an accepted security level for particular applications.

For interested readers, the recent survey of cryptography algorithms for IoT objects can be found in [85].

- **Hardware-based solutions:** One of the most effective countermeasures against Trojans, physical, and side-channel attacks is changing the circuit [35]. In what follows, we briefly illustrate how certain circuit modifications and changes may mitigate these attacks.

1. Minimizing information leakage: To address side-channel attacks, several techniques have been proposed, the most popular of which are shielding [59], adding randomized delay [86], deliberately-generated noise [87], and enhancing the cache architecture [88].

2. Tamper proofing and self-destruction: To enhance protection against physical attacks, IoT objects may be equipped with hardware-based solutions. For instance, many tamper-proofing techniques attached to physical packages of IoT objects have been proposed to prevent tampering against sensors. Furthermore, self-destruction methods can be used as an alternative technique to alleviate physical attacks [89].

3. Integrating Physically Unclonable Function (PUF) into the circuit: The process in which a noisy function is added to an integrated circuit is known as a PUF. Having queried with a challenge z , a PUF creates a reply y that relies on both z and a unique built-in feature of the object [90]. PUFs are supposed to be physically unclonable, and tamper-proof [91]. Moreover, PUFs provide unique object identification and authentication as well as Trojan detection methods by detecting unintended changes in the circuit [51].

It is worth noting that the previously-mentioned PUF techniques are designed for nodes, whereas RFID tags have their own techniques to prevent both counterfeiting and cloning attacks by integrating PUFs into RFID tags [92,93].

4. Run-time attestation: Another effective countermeasure used to mitigate physical and side-channel attacks is known as a run-time attestation in which an object can generate a proof about its firmware attested by a remote entity [94]. To this point, several approaches have been proposed in the state-of-the-art, such as TMP [95] and TrustZone [96].

- **Securing firmware update:** There are two methods in which update firmware can be provided: (i) a remote update and (ii) a direct update. In the situation of a remote firmware update, a command to advertise the availability of a new version of firmware is broadcast by a base node(server). Then, a node with a new updated version propagates an announcement to contiguous nodes. Upon receiving such update, the nodes willing to update their firmware will compare their existing firmware with new one and send requests if they require an update. Finally, the advertiser begins transmitting data to the requesters. To provide a secure approach for nodes updating their firmware remotely, all the requests, responses, and data packets should be authenticated and encrypted. Furthermore, during each step of this complex process, threats posed by Dos attack should be addressed with caution [97]. In the case of a direct firmware update (e.g., using USB), an end user attempting to install firmware should be authenticated.
- **Intrusion Detection system:** IDSs will be discussed later on when we investigate proper security measures for IoT communication. However, in this section, we focus only on policy-based IDSs for addressing security and privacy guidelines suggested at this layer. IDSs ensure that general policies are not violated by a continuous-monitoring process. More importantly, they provide a reliable method to mitigate both battery-draining and sleep deprivation attacks by observing abnormal requests to objects. Many up-to-date and ongoing research works have been proposed to monitor edge nodes and mitigate possible attacks at this level [98,99]. For interested readers, recent surveys published in this topic can be found in [100–103].
- **Decommissioning:** As all good things must come to an end, IoT objects will reach a point in which they must be de-provisioned; these objects need to be removed and cannot be returned back to the network [104]. In spite of the importance of decommission for solving some security and privacy issues (e.g., personal data breaches), there is a lack of research efforts conducted in literature in this regard, let alone its implementation. However, Smart Card Alliance in [39] has suggested two choices for decommissioning. Firstly, the objects can be reset to factory default mode. In this option, all data in such objects will be deleted except the basic security parameters. These objects can come back to life later. Secondly, a blacklist technique implemented on a server will be used to prevent blocked objects to re-join an network unless their statuses on the server have been changed.
- **Isolation:** One of the most effective techniques to protect privacy of RFID tags is to isolate them from EM waves. One approach is to construct and use separation rooms. This solution, however, is very expensive [44]. An alternative solution is proposed in which an isolation container made of metal is used to hinder EM waves. Such container is known as a Faraday cage [105]. Another technique is proposed to impede specific radio channel using an active radio frequency jammer.
- **Blocking:** In [63], the authors have proposed an effective approach, called blocking, for protecting privacy of RFID tags. In such a technique, a modifiable bit known as a privacy bit is attached to

each tag. Setting a privacy bit to ‘0’ refers that public scanning of tag is possible, while setting a privacy bit to ‘1’ indicates that tag is private. This approach needs a certain kind of tags known as a blocker tag. Another approach known as a soft blocking proposes in [106]. It depends heavily on a reader’s configuration to compel a set of policies that is realized in a system. This set of policies ensures that readers read only public tags. Violating tag policies by a reader can be detected using a monitoring device.

- **Anonymous tag:** In [107], the authors propose a new approach based on look-up table mapping for protecting privacy of RFID tags. The main contribution of this work is to store a mapping between an anonymous ID and a genuine ID to prevent an attacker to discover the mapping schema to identify genuine ID from the anonymous one. In spite of emitting an anonymous ID by tag, an attacker can still track RFID if its ID is not changed over time. Therefore, the anonymous ID should be changed continuously to tackle the tacking issue [48].
- **Distance estimation:** Identifying the distance between a tag and a reader based on signal-to-noise ratio is suggested in [45]. The authors claim that it is potential to deduce a metric in which the distance of a reader trying to read a tag data is estimated. This allows the tag to only offer distance-based information. For instance, scanned at 10 m, the tag may publish public information but provides its unique identifier with 1 m distance.
- **Trojan activation methods:** The main purpose of a Trojan activation method is to partially/fully enable the Trojan circuitry in which hardware Trojan is detected. Many Trojan activation techniques have been suggested [50,108]. The common objective of such techniques is to detect the differences between a Trojan-free circuit and a Trojan-inserted circuit. For instance, the authors in [109], have proposed a new Trojan activation method known as MERO. Its main function is to extract a compressed set of test patterns, and at the same time it provides a full coverage of Trojan detection.
- **Customer responsibilities:** In spite of the importance of the above-mentioned countermeasures to alleviate IoT attacks and threats, the customer has an indispensable role for preventing some IoT attacks. For instance, changing default passwords for IoT objects lies on customer’s shoulders to prevent Dos attacks. In [30], the authors describe some IoT malwares like Mirai, Carna, and BASHLITE which cause DDos attacks in IoT by exploiting default credentials, since most of IoT objects come with default passwords that are not changed by customers.

An overview of countermeasures proposed for edge node level can be found in Table 4.

Table 4. A summary of some implementation techniques proposed for edge nodes level.

Implementation Techniques	Research Proposal	Year	Mechanism Used
Hardware Trojan Detection	[64]	2015	Compares the Power consumption of a Trojan-free IC with a Trojan-inserted IC
	[66]	2014	Uses Test-vector selection approach and path-delay structure
	[67]	2014	Uses symmetry breaking in path delays
	[65]	2018	Uses off-the shelf techniques like power analysis report, thermal measures and side-channel analysis
Malicious Firmware Detection	[36]	2016	Uses dynamic permutation
	[71]	2014	Utilizes instruction-level power consumption templates to elicit data about finished instructions by the processor
	[73]	2016	Proposes a low-cost approach to identify malicious alterations in the firmware of objects by calculating hardware events happen during the operation of firmware
	[72]	2013	Verifies program’s behaviour using the object’s side-channel leakage
Encryption	[75]	2010	Proposes an anti-tamper JTAG used a true number generator and SHA256 secure hash to generate a challenge/response for IC test
	[76]	2017	Suggests a secure trace-based debugging schema
	[79]	2018	Proposes symmetric encryption technique for RFID applications based on dynamic generation of key
	[27]	2013	Proposes a secure JTAG interface based on public-key cryptography (ECC) which provides mutual authentication between the object and the server
Hash-based technique	[110]	2011	Proposes a Multi-level secure JTAG technique based the Access Monitor and Security Authentication Module
	[80]	2005	Proposes an authentication protocol (OHLCAP) for RFID which needs only one one-way hash function
	[80]	2005	Proposes an authentication protocol (LCAP) for RFID which needs only two one-way hash function
	[49]	2004	Introduces three different security approaches (Hash-based access control, randomized access, and backward channel key negotiation) for RFID

Table 4. Cont.

Implementation Techniques	Research Proposal	Year	Mechanism Used
Lightweight protocols	[83]	2006	Suggests a lightweight mutual authentication protocol (300 gates) for RFID tags
	[84]	2004	Suggests a tree-based scheme in which private authentication between tags and its reader is accomplished
Minimizing information leakage	[59]	2016	Proposes a new type of information security attacks that discloses privacy by taking advantage of information leakage like physiological
	[87]	2011	Introduces a novel mechanism to prevent differential power analysis on smart cards and ICs by generating a high level of noise
Integrating PUF into the circuit	[90]	2010	Proposes a new class of sensor (node) which enlarges the functionality of PUFs to offer verification, unclonability, and authentication
	[111]	2013	Proposes an PUF-based authentication protocol to prevent cloning attacks
	[93]	2014	Proposes an PUF-based authentication protocol to prevent memory leakage
Run-time attestation	[94]	2018	Introduces a technique based on device identifier composition engine to securely create attestation indication at runtime utilizing CPU characteristics (e.g., Memory Protection Unit)
	[96]	2005	Proposes a TrustZone hardware technique that offers a security framework allowing an object to mitigate several threats (e.g., unauthorized access to JTAG)
	[95]	2011	Proposes a hardware-based technique that integrates the Trusted Platform Module (TPM) into a processor.
Intrusion Detection system	[99]	2012	Proposes an efficient algorithm based on Markov model for sensor nodes to detect abnormal activities (e.g., Dos attacks)
	[98]	2012	Proposes an efficient algorithm based on Markov model for sensor nodes to detect abnormal activities (e.g., Dos attacks)
Decommissioning	[39]	2016	Suggests two ways in which objects can be decommissioned (blacklist and reset to factory default mode)
isolation	[44]	2008	Suggests an isolated container made of a metal mesh to protect privacy of tags
	[105]	2002	Suggests jamming all neighboring radio channels by an active RF jammer that frequently hinders particular RF channels
Blocking	[63]	2003	Suggests the use of blocker tags as a technique for preserving users' privacy as result of integrating RFID tags into their products
	[106]	2004	Proposes a new version of blocker concept called a soft blocking that provides flexible privacy policies
Anonymous tag	[107]	2009	Suggests a look up mapping mechanism for achieving the goal of location privacy by converting genuine ID of RFID into anonymous one.
	[48]	2006	Proposes a novel technique to change an anonymous ID frequently to prevent some privacy issues (e.g., tracking attack)
Trojan activation methods	[108]	2015	Proposes hardware Trojan detection that uses a comprehensive testing of k-bit sub-spaces of signals
	[50]	2015	Proposes a Trojan detection technique based on probability obfuscation scan chain
	[109]	2009	Proving to be very efficient in discovering the existence of Trojan in an IC

4. Level 2: Communication

In this section, we first introduce a comprehensive IoT security and privacy guidelines for IoT communication. Then, we investigate potential attacks and threats at this level. Finally, we recognize proper countermeasures found in literature to implement our proposed guidelines and mitigate possible attacks at this level. Figure 4 summarizes the relationship between proposed guidelines, their suitable countermeasures, and attacks at this level.

4.1. Security and Privacy Guidelines for Communication Level

Next, we describe required security and privacy guidelines suggested for IoT communication.

- **Avoid proprietary protocols:** In [112], the authors state that several IoT protocols have been developed for securing IoT communication, but they are not standardized, since they are designed for specific applications. Such protocols may hinder interoperability and introduce new security threats, as their security mechanisms have not been tested in a large scale.
- **Reduce Interference:** Most IoT objects are equipped with connectivity technologies, among which wireless protocols are the most common ones. These protocols are susceptible to interference. For instance, an attacker can easily send signals in which communication link (e.g., wireless link) can be interfered, preventing the delivery of packets or messages [113].
- **Enable security mode:** This guideline suggests that an IoT protocol, if it has different security modes, operated at any layer of IoT stack, should always enable its security mode by default [32]. There are many IoT protocols which have different security modes such as RPL, CoAP, 6lowpan,

and Bluetooth Low Energy. For instance, CoAP has been designed with four security modes, namely Nosed, PreSharedKey (PSK), RawPublicKey (RPK), and certificate.

For interested readers, the most recent surveys published in this regard (IoT security protocols) can be found in [112,114,115].

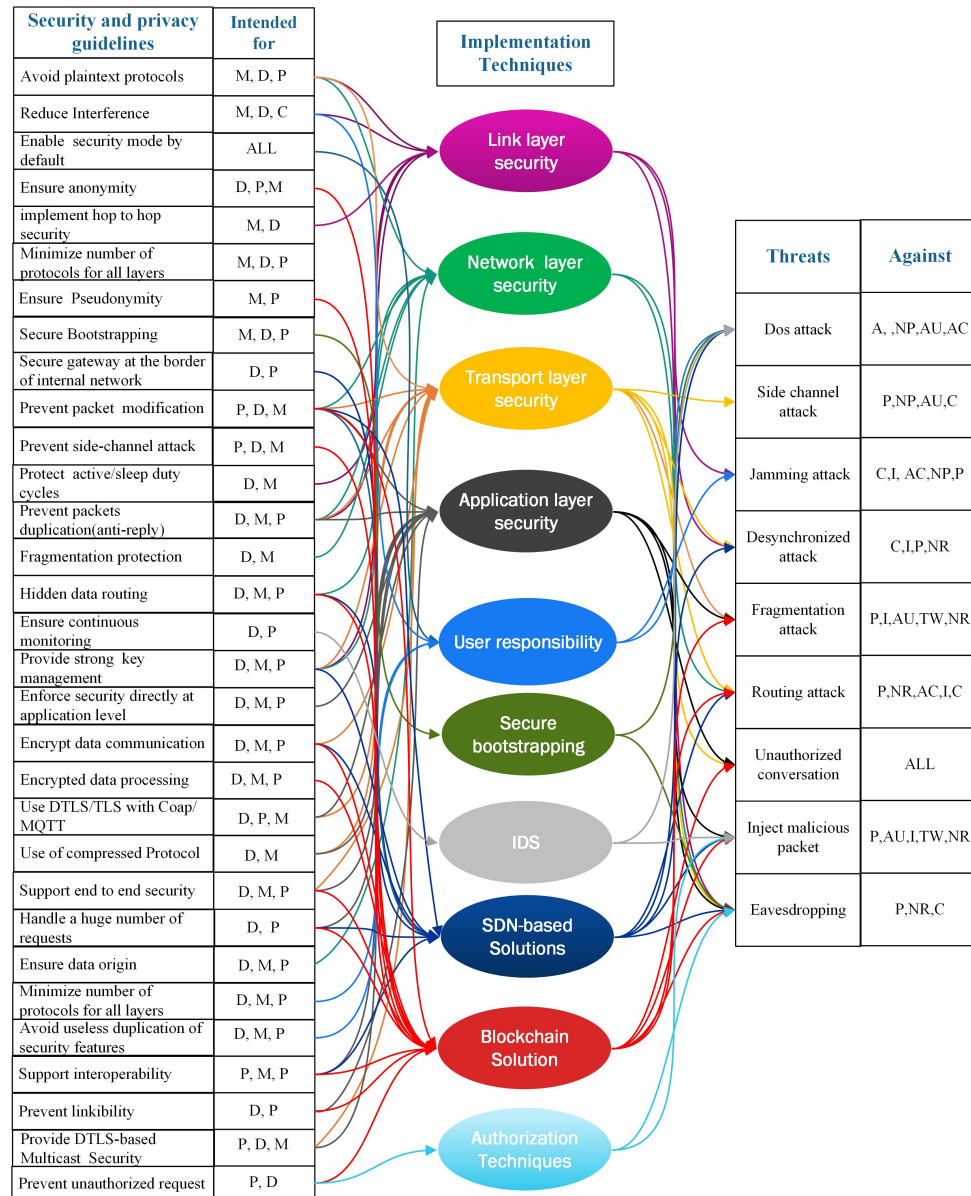


Figure 4. An overview of guidelines, stakeholders, attacks and countermeasures for communication level.

- Prevent packet modification:** In remote health monitoring systems, modifying packets or inserting invalid ones may cause harmful consequences (e.g., loss of human lives). To prevent a message modification by an attacker, this guideline suggests that IoT protocols should be equipped with message integrity techniques [112]. The authors in [116], for example, propose a lightweight protocol used to secure video streaming. Such protocol achieves several security goals, among which is data integrity.
- Ensure anonymity:** Another key privacy requirement of IoT communication is anonymity in which personally identifiable information should be removed from packets [117]. For instance, the authors in [118] propose an effective approach in which anonymity of IoT communications is accomplished.

- **Ensure packet origin authentication:** In [119], the authors have stated that packets origin authentication can be used to prevent the injection of malicious packets by an attacker to an IoT system. Therefore, this guideline suggests that data origin authentication must be integrated into IoT protocols. Examples of integrating packet or data origin authentication into IoT protocols can be found in [116,120].
- **Network Segmentation:** It is an effective technique used to prevent the propagation of IoT attacks by dividing an IoT network into several segments based on different policies (e.g., protocols). In this case, if one object is compromised, other objects will not get hacked, since they are located in different segments [121]. For example, a wireless network used Service Set Identifiers (SSID) can be divided into multiple SSIDs, instead of one [112].
- **Minimize number of protocols for each layer:** As mentioned above, minimizing hardware components will reduce the IoT attack surface. Similarly, minimize number of IoT protocols at each layer will not only reduce the attack surface, but also minimize the used resources [2].
- **Implement hop-to-hop security:** It means that each object can decrypt and encrypt the messages to the next hop using shared keys. Hence, the content of each message is known by each hop. Despite disclosure of the content of messages, there are some situations in which IoT objects should only implement hop-to-hop security. One situation is when IoT objects have very limited resources, preventing them from implementing end-to-end security. In this case, such objects can use secure link-layer protocols (e.g., IEEE 802.15.4) to implement hop-to-hop security [117].
- **Handle a huge number of requests:** In [122], authors have shown that some IoT objects may encounter similar impacts as Dos attacks from a huge amount of valid requests coming from different objects, making IoT services inaccessible. So, this guideline suggests that IoT protocols, particularly application layer protocols, should have capabilities to handle effectively a large number of requests.
- **Secure Bootstrapping:** In IoT, a bootstrap is a procedure via which an IoT object is connected either to another IoT object or to a network. The importance of equipping an IoT object with a secure bootstrapping schema stems from exchanging a number of settings securely between IoT objects, such as link-layer encryption keys, network names, and wireless channels [123]. Bootstrap is also used to authenticate a new identity, authorize access to IoT network, and register a joining object. Its implementation depends heavily upon the architecture (centralized or distributed) [124]. In general if such guideline implemented, it will prevent eavesdropping, and unauthorized access attacks.
- **Prevent linkability:** In [125], the authors propose a new privacy requirement for IoT communication known as unlinkability. In such requirement linking data or events to a specific person is not allowed. For instance, the authors in [126] have shown that a retailer company in the U.S. known as Target once got complaints from a client who was really upset and disappointed, since such company gave coupons for infant clothes to his youth daughter. That said, it is worth noting that Target deliberately sent such coupons to the daughter because she was pregnant at that period. This kind of inference may take place as a result of storing data with its personally identifiable information or performing data mining on its clients' data.
- **Ensure Pseudonym:** In [125], the authors also propose another privacy requirement for IoT communication known as Pseudonym in which data is linked to actions, instead of a person.
- **Prevent packets duplication:** Providing packets or messages freshness is an indispensable security requirement for IoT protocols. This guideline, therefore, suggests that each IoT protocol should have a mechanism in which duplicate packets are discarded or prevented [1].
- **Protect active/sleep duty cycle:** Being equipped with tiny batteries, IoT objects should have duty cycle mechanisms to reduce battery consumption by switching the radio off as much as possible. Therefore, several types of duty cycle protocols have been proposed (e.g., IEEE 802.15.4) [127]. Security goals of an object could be compromised if an attacker find a way to drain its battery.

For example, an attacker could deactivate a fire detection system by sending tons of requests to deplete its battery by keeping its radio active for a long time [54].

Recently, the authors in [128] investigated the cost of IoT security on duty cycle protocols.

- **Hidden data routing:** Upon its arrival at cloud servers, IoT data is expected to move between several objects. Due to this movement, an attacker could reveal customer's location using traffic analysis techniques. To mitigate such a threat, this guideline suggests that IoT protocols should be equipped with anonymous routing techniques, for instance Tor [129].
- **Provide a strong Key management:** To realize full potential of IoT, its security and privacy issues need to be addressed, the most important of which is a key management. Its importance stems from its involvement roughly in all IoT security mechanisms [130].
- **Enforce security directly at application level:** Despite the benefits of an end-to-end security to transfer packets securely, its implementation is really hard when a communication link depends on many intermediate objects working at proxies. This guideline, therefore, suggests providing security at application layer [2].
- **Encrypt data communication:** To prevent unauthorized access to critical information (e.g., passwords, object identifier, and object configuration) during conversation between IoT objects, this guideline suggests that a communication link should be encrypted. It is worth mentioning that IoT paradigm has four communication patterns: (1) object-to-object, (2) object-to-gateway, (3) object-to-cloud, and (4) gateway-to-cloud.
- **Encrypted data processing:** Although encryption can be used to prevent unauthorized objects to read packets, objects that process data can read it. To prevent them from doing so, this guideline suggests that data processing should be carried out while data is encrypted [13].
- **Use DTLS with CoAP and MQTT:** CoAp stands for Constrained Application Protocol developed by an IETF working group known as RESTful Environments [131]. It is a lightweight application protocol used to exchange messages between IoT objects or between an object and a resource-rich object. As CoAp itself does not offer any security mechanisms for data protection, nor does it provide authentication, this guideline suggests that DTLS at transport layer should be integrated with CoAP to secure its packets [132].

MQTT stands for Message Queue Telemetry Transport developed in 1999 and standardized in 2013. It is a lightweight application protocol used to exchange messages in many IoT domains (e.g., energy, health, and etc.). It consists of publisher, subscriber, and broker. From a security perspective, each MQTT message has a changeable header (longer than 2 bytes) containing a password and a user-name for authentication purpose in which a broker can deny unauthorized connections. That said, such connections to broker are insecure, since passwords and usernames are unencrypted. This guideline, therefore, suggests that a secure version of MQTT which equips security mechanisms (e.g., proposed in [133]) should always be used.

It is worth noting that MQTT has an extension known as a SMQTT to solve security issues, but it is not complete [133]. More information about MQTT as well as CoAP can be found in [112].

- **Provide DTLS-based Multicast Security:** Several research proposals have integrated CoAP with DTLS so that an end-to-end secure communication will be provided [7,8]. However, DTLS lacks group key management which hinders implementation of multicast communication in CoAP [134]. Therefore, the authors in [135] propose a new schema in which mutlicast group communication for CoAP objects is enabled by DTLS record protocol.
- **Support end-to-end security:** Although hop-to-hop security transfers securely packet to its destination using different hops, each participating hop is able to read the content of the packet. Therefore, this guideline suggests that end-to-end security should be implemented in IoT protocols [136].

- **Support interoperability:** Another key concern regarding IoT security protocols is an interoperability, allowing negotiation of security parameters to be utilized during operations. These negotiations may be associated with digital signature and cryptographic techniques. To assure a full interoperability among IoT objects, defining a set of compulsory choices that should implement in all IoT objects, is required to provide minimal support [2].
- **Use of compressed protocols:** A protocol compression is one of the main security issues regarding IoT protocols, as it plays a vital role in reducing the necessity for message fragmentation techniques which delay transmission.
- **Avoid useless repetition of security features:** This guideline suggests preventing pointless duplication of security parameters at each layer of the IoT stack. Such repetition may lead to severe influences on computation performance as well as transmission, for instance, using both IPSec and TLS/DTLS to achieve end-to-end security [2].
- **Prevent unauthorized requests:** This guideline suggests that each IoT object should be shielded with authorization techniques (e.g., a role-based technique proposed in [137]) via which all unauthorized requests can be blocked or ignored.

4.2. Attacks and Threats against Communication Level

Next, we describe attacks and threats against IoT communication:

- **Side channel attacks:** Despite the difficulty to implement such attack at this layer, it is a strong attack against encryption techniques which may affect their security and reliability. Unlike side-channel attacks, launched at edge node level, side-channel attacks at this level are not invasive, since they only elicit intentionally-leaked information. This type of attacks at this level are undetectable, and thus they are very hard to defend them. However, adding noise and minimizing leakage can be used to lessen them.
- **Collision attacks:** This type of attacks can be launched on the link layer. One scenario is that intentionally-generated noise against communication links (e.g., an IEEE 802.15.4) can be used by an adversary to create a collision [114]. This collision needs re-transfer of packets affected by the collision. Using such method an attacker can easily drain object's battery by creating many collisions, resulting in too many re-transmissions [138].
- **Fragmentation attacks:** Although 6LoWPAN lacks any security mechanisms, its security is offered by underlying layers (e.g., an IEEE 802.15.4). The IEEE 802.15.4 has Maximum Transmission Unit (MTU) of 127 bytes, whereas IPv6 has a minimum MTU of 1280 bytes. Being developed with fragmentation technique, 6LoWPAN provides the transfer of IPv6 packets over IEEE 802.15.4. In this case, an attacker can insert a malicious packet among other fragments, as 6LoWPAN has designed without authentication techniques [117].
- **Routing attacks:** Attacks targeting how packets are directed are known as routing attacks. The main impacts of such attacks at this level can include, but not limited to, misrouting, spoofing, and dropping packets. Modifying the routing information, so called modifying attack, is the easiest form of routing attacks. Besides these attacks, many attacks have been found in literature, the most common of which are Sybil [139], Gray Hole [140], Wormhole, Hello flood, and Selective forwarding. Next, We briefly illustrate them.

1. Hello Flood: Having used a malicious object with great transmission power, an attacker can propagated "HELLO PACKETS" to announce maliciously his/her existence to the whole network [117].

2. Gray Hole: It is a special kind of Black Hole attacks. In such attack, IoT objects may discard some packets. For interested readers, the recent survey published in relation to this topic can be found in [140].

3. **Sybil attack:** In this type of attacks, an attacker can claim different identities to out-vote honest nodes.
 4. **Worm Hole:** In worm hole attacks, an attacker could establish a tunnel between two malicious objects to transfer packets to the wrong destination.
 5. **Selective forwarding:** In this attack, an attacker can either refuse to forward packets or transfer only specific packets to disturb routing paths. It is worth noting that in [117], the authors present many other severe attacks, which are not presented above.
- **Eavesdropping:** Deliberately listening to private messages or packets over communication channel is known as an eavesdropping. It is an effective attack against communication links If messages are not encrypted during a conversation. In this case, an attacker can extract valuable information (e.g., usernames and passwords). Threats posed by eavesdropping may significantly increase when packets convey access control information (e.g., object identifier, object configuration, and shared key).
 - **Inject malicious packets:** There are three ways in which an attacker may insert malicious packets into communication channels, namely insertion, replication, and alteration [138]. In insertion attacks, an attacker can generate malicious packets, which seem legitimate, and insert them into a communication link. Alteration attacks depend heavily on capturing, modifying the packet (e.g., checksum and data), and then sending the modified packet. In alteration attacks, previously-exchanged packets will be captured and sent again to the network.
 - **Unauthorized conversation:** To share and access data, each IoT object requires to communicate with other objects. That said, each object must only interact with a set of objects which need its data. This kind of restricted interactions will prevent unauthorized access to IoT objects which is a fundamental security requirement of IoT. For instance, a thermostat, in a smart home, depends heavily on a smoke detector's data to turn a heating system off in case of danger. Nevertheless, insecurely sharing data with other objects by the smoke detector may put the entire smart home at risk [35].
 - **Dos attacks:** Jamming attacks at communication level which prevent transmission of packets are the most common types of Dos attacks. Two kinds of such attacks have been identified in the existing state-of-the-art, namely constant jamming and Sporadic jamming. A constant jamming causes a complete interference of the entire network, preventing objects from sending/receiving packets. In contrast, a sporadic jamming causes a partial interference of the network, allowing objects to send or receive packets intermittently [117]. For instance, an attacker can prevent a fire detection system to notify a fire department in case of an emergency by jamming its communication channel [35].

For interested readers, the recent survey published in relation to this topic can be found in [141].

- **Desynchronized attack:** In this type of attacks, an adversary can impede an active connection between two objects by sending forged packets with a fake sequence number, resulting in desynchronization between two objects. Such desynchronization will require endpoints to retransmit already sent packets [138].

4.3. Implementation Techniques Suitable Communication Level

Next, we describe countermeasures for implementing our proposed security and privacy guidelines and mitigating possible attacks against communication level.

- **Secure Bootstrapping Techniques** According to [124], the implementation of secure bootstrapping techniques depends heavily on their architectures either distributed or centralized. In a distributed architecture, two IoT objects can reach an agreement on a common secret using a Diffie-Hellman algorithm. In general, performing a key exchange and setup of security parameters without

a trusted party can be achieved using several protocols, such as TLS, DTLS [142], Host Identity Protocol (HIP) [143], and IKEv2 [144]. That said, it is really difficult to implement such protocols on very constrained objects. To overcome this issue, various research efforts have been proposed such as Diet HIP [145] and human memorable password in which trust links between IoT objects and gateway are established [146].

In a centralized architecture, the distribution process of operational keys in any security domain depends entirely on a single object that can hold either certificates or predefined keys. The implementation of such architectures in IoT have been investigated by many researchers. For instance, the authors in [123] suggest the use of the Protocol for Carrying Authentication for Network Access (PANA), for conveying of the Extensible Authentication Protocol (EAP) messages between a PANA agent and a PANA client.

For interested readers, the latest survey published in this topic can be found in [147].

- **Adding security at link layer:** The IP-based communication in IoT objects depends heavily on 6LoWPAN [148], which in turn depends on the IEEE 802.15.4 link layer. It achieves different security goals like confidentiality, and integrity [134]. The IEEE 802.15.4 link-layer offers hop-to-hop security where each object in the communication link should be trusted without any authentication, key management, time-synchronized communications, and reply protection. To cope with the lack of reply protection as well as time-synchronized communication, a new extension (modification) of the IEEE 802.15.4 was introduced in 2012 by IETF called IEEE 802.15.4e [149].

It is essential to know that link layer security cannot protect packets once they have left its network. To tackle this issue, many security solutions have been proposed. In [150], the authors propose a key management system for wireless sensor networks. Such system adds security at link layer. In [151], ArchRock PhyNET uses IPsec in a tunnel model to secure a link between a border router and nodes. In [152], the authors propose a new keying mechanism used directly on a media access control.

- **Adding security at the transport layer:** Both TLS and SSL can offer end-to-end security [153]. Both techniques have been used widely to secure communications in the traditional internet, since they provide authentication, key exchange mechanisms, confidentiality, and integrity. That said, TLS and SSL cannot be used directly for IoT due to two reasons. First, TLS uses over TCP which is not the suitable approach for IoT objects because of their limited resources. Second, TLS/SSL session setup and keys exchange need a set of packets exchanges.

However, SSL and TLS have been proposed as security solutions for IoT. In [5], the authors have proposed a security mechanism for smart objects based on SSL. According to their evaluation, a full SSL handshake as well as packets exchanges needs 2 seconds to finish.

In [6], the authors propose a lightweight TLS protocol in which a secure communication between smart nodes and a remote terminal are achieved. As this solution depends heavily on the border router to minimize computational efforts on tiny nodes, it cannot achieve a full end-to-end security solution.

Datagram Transport Layer Security (DTLS) introduces to offer security goals similar to TLS, but it builds over UDP. To this end, several solutions have been proposed in the literature that provide end-to-end security. In [7], the authors introduce a new architecture for integrated sensing applications. Such architecture provides an end-to-end transport layer security and a mutual authentication using Elliptic-curve cryptography (ECC). In [8], authors propose two approaches (HTTP/TLS and CoAP/DTLS) to provide end-to-end security between two objects situated in homogeneous networks by translating between DTLS and TLS. In [9], the authors propose a fully-implemented two-way authentication approach for IoT objects. This approach depends heavily on current Internet standard, notably DTLS protocol. The exchange of x.509 certificates

that contain RSA keys and authenticated DTLS handshake has been utilized to implement this technique.

Several other research proposals have been introduced in this regard [136,154].

- **Adding security at network layer:** Next, we discuss the research proposals which provide solutions to protect network layer communications using 6LoWPAN and RPL.

1. Adding security to 6LoWPAN: 6LoWPAN stands for IPv6 over Low power Wireless Personal Area Networks [148]. 6LoWPAN is a network layer protocol standardized by IETF. Having equipped with a header compression mechanism, it provides internet connectivity on resource-constrained objects.

As 6LoWPAN does not provide security techniques, nor does it offer key management, several research efforts discussed below have been proposed in this regard.

In [155], authors introduce novel compressed security headers suitable for 6LoWPAN to provide end-to-end network layer security. Such security headers simplify the integration of 6LoWPAN with IP Security architecture.

In [156], the authors propose an IPsec extension which is suitable for 6LoWPAN to offer security for IoT objects based on IPsec technique. Unlike link layer security, 6LoWPAN/IPsec is a candidate solution for securing IoT objects in terms of energy consumption, processing time, and packet size. It also performs better than link layer security, when data size and the number of objects increase.

In [157], the authors have proposed a lightweight IKE protocol suitable for resource-constrained objects by compressing IKE headers at 6LoWPAN layer.

Several other research proposals have been introduced to this end [158–161].

2. Adding security to RPL: RPL stands for IPv6 Routing Protocol for Low-Power and Lossy Networks. RPL is a network layer protocol which standardizes by IETF [162]. Such protocol identifies the method in which routing is carried out inside Low-power and Lossy Networks (LLNs). It also describes the RPL packets transmitted between LLN objects over ICMPv6. These packets form routing table inside the LLN. The RPL specification defines three security techniques: unsecured, authenticated, and preinstalled. However, other security mechanisms (e.g., protection against internal attacks) are required to support its operation [163].

Next, we discuss recent research efforts addressing security for RPL.

Although RPL specification provides security mechanism against external attack, it does not offer security solution against internal attacks [162]. To cope with this issue, the authors in [164] investigate several types of internal attacks on RPL targeting particularly RPL rank which provides several benefits, such as route optimization, management of protocol overhead, and prevention of the loop. The authors also analyze the consequences of these attacks on the performance of the network. Such attacks stem from RPL vulnerability, which is its inability to validate services offered by the parent. In [165], the authors also address internal attacks against RPL. More particularly, they focus on compromising an object to imitate the Destination Oriented Directed Acyclic Graph (DODAG) root (gateway) by an internal adversary. To mitigate such attacks (e.g., a malicious increase of rank value), the authors introduce a new security technique known as a Version number and Rank Authentication security (VeRA). Such technique combines version numbers with Message Authentication Code (MAC) and signatures. In [166], the authors investigated the consequences of a sinkhole attack. Moreover, the authors assessed two defense mechanisms, namely a parent fail-over and a rank authentication, via which sinkhole attacks can be mitigated.

The interested readers can have a look at the latest surveys published in this topic [121,167].

- **Adding security at application layer:** Next, we discuss research proposals focusing on integrating DTLS with CoAP.

Despite the benefit of using DTLS to provide end-to-end IoT security, it has some drawbacks which need to be investigated to ease its integration with CoAP. One important limitation is that the DTLS lacks key management mechanisms, preventing multicast group communications in CoAP [134]. To this end, the authors in [135] adapt DTLS record protocol in such a way that multicast group communications in CoAP objects are carried out and protected. Another limitation is that DTLS handshake can have a direct impact on resource-constrained objects. Therefore, the author in [168] investigates several issues that may hinder the implementation of DTLS in resource-constrained objects, for instance, the stateless compression of DTLS headers to minimize the overhead of handshake and record protocols. Many other research proposals have attempted to simplify the integration of DTLS with CoAP to support constrained objects [132,169,170].

Besides previously-discussed issues related to enhance security at the application layer using CoAP, there are some popular problems addressed by several research works including, the lack of mapping techniques between TLS and DTLS investigated in [9], the absence of digital Certificate and Public-Keys investigated in [154], and most importantly the enforcement of object security with CoAP addressed in [171,172].

- **Intrusion detection systems (IDS):** In spite of adding security mechanisms at each layer of IoT stack, IDS is fundamentally required as a second line of protection in which communication links as well as network operations are monitored. Moreover, it can detect abnormal activities. For instance, IDS can detect any violation of pre-defined rules. As several conventional IDS proposals have been adapted for WSNs [173,174], a few modern IDS approaches have been investigated IoT security and privacy issues directly. To this end, the authors in [175] propose a novel IDS for IoT objects called SVELTE, which is the first IDS developed to match the requirements of IPv6-enabled objects. Such IDS is capable of detecting several routing attacks like Black Hole, selective-forwarding, and sinkhole. In [176], the authors propose another IDS based on artificial immune technique composed of a set of detectors (e.g., memory, mature, and immature) and the attack information library, both of which can be used to detect abnormal activity in IoT environment.

For interested readers, the latest survey published in this topic can be found in [100].

- **Blockchain-based solutions:** Blockchain is an emerging technology that has shaped the universe of cryptocurrency (e.g., bitcoin), aiming to construct transactions or communications between objects in a distributed architecture without the need for centralized trust entities. Furthermore, a trust model between objects is not required. In such technology, once the transaction is validated, it is impossible to deny it. In addition to its use in cryptocurrency, several researchers have begun to shed the light on such technology to address different IoT security and privacy issues. One important issue that has been investigated in literature is secure IoT transactions. Several proposals have been conducted to tackle this an issue. Some examples of them are HTTPS protocol for IoT objects [177], a multi-layer security architecture for smart cities [178], a decentralized key management for IoT objects [179], a platform for industrial Internet of Things [180], and trsutchain-based transactions [181]. Data sharing is another key issue that has been investigated by different proposals, example of which are a decentralized approach for sharing data in IoT [182] and a healthcare data sharing [183].

In spite of the blockchain's benefits mentioned-above, its adaptation into IoT faces different challenges required to be solved including powerful computational capabilities investigated in [184], bandwidth consumption, not full anonymity, and more importantly time latency (e.g., 10 min per transaction) [185].

For interested readers, a recent survey about the integration of blockchain into IoT along with challenges and opportunities can be found in [186].

- **SDN-based solutions:** SDN is a new emerging technology revolutionized the universe of networks. The separation between network control plan and data plan is the main objective behind such technology, initiated in 2011. This kind of separation would allow a dynamic management of network, a centralized configuration, and control of the network [187]. In SDN paradigm, objects (e.g., routers, gateways, and switches) cannot perform control decisions (e.g., forwarding tables), but they can learn such decisions from a centralized entity known as a SDN controller [188]. Due to a centralized architecture, SDN is an effective technique to address some IoT security challenges. Several SDN-based proposals that investigate security concerns (e.g., management of security policies, identifying abnormal activities in the network, and prevention of Dos attacks) in IoT can be found in [189–191].

Although SDN technology can be used to address some IoT security issues mentioned above, it has some drawbacks including, but not limited to, operating in centralized architectures, lack of scalability, and most importantly not suitable for dynamic environments [187].

- **Authorization solutions:** In order to restrict system access to authorized requests, authorization mechanisms must be taken into account when developing IoT systems [2]. Authorization techniques must verify if two objects participated in communication have been validated. The most common authentication techniques are a role-based access control (RBAC) and an attribute-based access control (ABAC). ABAC converts privileges to a set of attributes assigned to an object, whereas RBAC converts privileges to a set of roles assigned to an object. Another technique which can be used to ensure authorization for IoT objects is known as Authentication and Authorization for Constrained Environments (ACE) [192]. Several research proposals related to ACE can be found in [192–195].
- **Customer responsibility:** Despite the previously-mentioned security mechanisms, there is always space for IoT customers to improve IoT security and reduce attacks and threats in IoT communication. For instance, it is a customer's responsibility to ensure that their objects are updated from time to time.

5. Digital Rights Management in IoT

In general, the need for digital rights management (DRM) for connected objects stems from legal access to their digital contents. This is because copyright holders want to fight piracy via the use of DRM systems developed to be gradually more difficult to breach. This kind of protections, however, would bring increased limitations that could restrict the ability of customers to buy content in methods allowed under fair use. For instance, the authors in [196] stated that code-based limitations applied in DRM techniques give copyright holders the capability to restrict the fair use rights further than permitted under copyright legislation. Apart from the limitations in fair use rights, customers are also subjected to lose their privacy because of DRM techniques which can be used to gather, share, and store customers data. Such data includes, but not limited to, personal data (e.g., contact lists), system configurations, and location data. Furthermore, some companies, for analyzing purpose, may also collect other information like customers IDs, gender, and IP addresses [196,197].

DRM systems depend heavily on access control technologies utilized to limit the use, distribution, and alteration of secured digital right contents. Different access control techniques like user authentication and user identity verification play a key role in determining the success of a DRM system [198]. To this point, several research proposals have conducted to provide secure DRMs for connected objects like mobile objects [198–201] and IoT objects [202]. For instance, the authors in [202] propose a set of requirements which can be used to enable automatic rights management service on IoT.

6. The Challenges of IoT Security Guidelines

In this section, we identify the following obstacles to defining, implementing and analyzing security and privacy guidelines for IoT.

6.1. Limited Device Resources

Implementing traditional security and privacy guidelines in IoT may necessitate a considerable re-engineering. This is because the majority of IoT objects have limited capabilities (e.g., memory, processing, and power). Therefore, there is a need for security and privacy guidelines proposed specifically for IoT system. That said, a careful implementation of such guidelines is of paramount importance to avoid useless duplication of security features which may degrade system performance [2]. For example, one guideline may suggest to use end-to-end secure communication. In IoT stack, composed of five layers (e.g., transport, network, and application layers), this guideline can be accomplished in three layers: (i) network layer by adapting IPsec, (ii) transport layer using TLS/DTLS, and (iii) application layer using object security. Combining these three security mechanisms causes severe impacts on computation performance as well as packets transmissions, since encryption and decryption process should be carried out in each layer.

6.2. Complex Ecosystem

As the IoT paradigm is enabled by several technologies (e.g., wireless protocols, RFID, sensors, and communication protocols), a set of security and privacy guidelines is needed to prevent them from being hacked. Nevertheless, security and privacy guidelines of IoT neither can be absolute nor guaranteed for two main reasons. First, weaknesses are frequently being exposed. In this case, there is an urgent need to observe, review and maintain security and privacy guidelines and security best practices designed for specific use cases and environments on regular basis. Second, as new technologies come in, new security and privacy guidelines are required, and unfortunately attackers will advance both their hacking tools and their level of knowledge to break them.

6.3. Privacy's Contextuality

Defining a set of privacy guidelines for IoT objects is a challenge, as privacy is a subjective term, and reaching an obvious and universal definition is extremely hard [203]. This is because it is hard to decide what data must be preserved (e.g., what may be important to a person may not be important to another person), when to preserve it, and to whom it must be available.

6.4. Never-Ending Process

IoT Security as well as privacy is a never-ending process and requires the involvement of all IoT stakeholders. An IoT system must remain secure during its life cycle. To cope with this issue, a continuous process to address its weaknesses is required by all stockholders, starting from manufacturers ending with customers. Manufacturers, for example, should equip their IoT products with updated mechanisms in which recently discovered vulnerabilities will be addressed. Moreover, manufactures should provide a clear end-of-life strategy for each product so that it can be destroyed without revealing its sensitive data. Such a strategy would contain valuable information including, but not limited to, how long will the update mechanisms be available, what is the default age of the product, when the product should be destroyed, and what is the required process for transferring the ownership of the product securely. On the other hand, several responsibilities for securing IoT objects lie on the shoulders of IoT customers like changing default passwords, setting a complicated unique password for each object, enabling security features (e.g., secure JTAG and TLS) if provided, and disabling remote administrative capabilities if not used. Besides the responsibilities of manufacturers and customers of securing IoT object, governments should also play a key role in facilitating better security and privacy by elucidating how current data protection, privacy, and customer protection laws will be

integrated with IoT. Furthermore, governments should force companies to apply such laws and prevent misleading representations of objects' security by companies.

7. Concluding Remarks

7.1. Recommendations for Future Work

In spite of considerable research efforts devoted to the IoT security domain, we can still suggest many issues that require to be addressed.

7.1.1. Lack of Awareness among IoT Stakeholders

The Lack of awareness regarding the benefits of security in IoT objects is common among all IoT stakeholders. Even worse, they lack the required knowledge in relation to attacks and the threats they may face in the future. For instance, the majority of IoT customers do not have a fundamental understanding of their IoT objects and the consequence on their environments. As a result, many IoT objects may not be updated and, thus, may be exposed to several attacks [14]. Manufacturers also should train and encourage their employees to follow security best practices [17]. Therefore, there is a need to increase the awareness among a new generation of IoT stakeholders (e.g., manufacturers and developers) about the impacts of the existing IoT attacks, the benefits of integrating security and privacy guidelines from the early stages of IoT development, and the use of proper countermeasures.

7.1.2. Lack of De/Commissioning Method

It is inevitable that each IoT object will reach its end-of-life stage and, thus, must be removed or disposed securely [104]. Despite the benefits of decommission for solving some security and privacy issues (e.g., personal data breaches), a few research efforts have been conducted in literature in this regard, let alone its implementation. The Smart Card Alliance in [39], however, suggested two choices for decommissioning: (i) a blacklist and (ii) a reset to factory default mode. In [185], the authors stated that block-chain technology can be used to solve both decommissioning/commissioning issues in IoT. A lot of efforts are needed to redesign such technology to be energy-efficient and lightweight in spite of requiring powerful computation.

7.1.3. Complement Cryptographic Algorithms With Context-Aware Approaches

Unlike other techniques, context-aware solutions are less developed in the existing literature in the context of IoT [187]. They could be used for complementing cryptographic techniques in order to operate efficiently. For example, if an object with limited resources is operating in a safe environment, it is not wise to equip such an object with heavy cryptographic techniques to perform an authentication process for other objects functioning in the same area. To this end, a lot of work is required to bridge the gap and improve the current IoT approaches by taking in account the environment in which objects are involved.

7.1.4. Firmware over The Air Need to Be Adopted for IoT

Although Firmware over The Air (FoTA) is used to update different rich-resource objects (e.g., tablets, phones, and etc.), it is not feasible for IoT objects, since these objects are resource constrained. However, Krishna et al. propose a lightweight FoTA process in which IoT objects can update their firmware securely using encryption and an object-signing certificate [204]. Further investigation is needed due to the lack of standardized approaches in the current state of the art.

7.1.5. Lightweight Cryptographic Protocols for Tiny Embedded Networks

The development of efficient cryptographic techniques with limited resources consumption like processing, energy and memory stems from the rapid increase in numbers of tiny embedded networks. With the advent of IoT, the scalability issue is increased due to such limited resources. The current cryptographic systems require energy, processing, and memory capabilities. Such capabilities may not

be available in embedded objects [3]. To cope with such issues, there is a need to develop a strong and efficient cryptography equipped with up-to-date energy producing mechanisms. To this end, several research proposals have demonstrated that elliptic curve cryptography can be used as a strong security technique because of limited resource consumption [205]. While other research proposals have shown that energy which might be produced from environmental situations of connected objects such as movement and vibration could be another solution to address limited resource consumption [206].

7.1.6. The Urgent Need of Standardized Security and Privacy Guidelines for IoT

As industries, IoT stakeholders, and technologists have distinguished the advantage of IoT in their daily lives like industrial revolution and automate processes within homes and companies, the adaptation of IoT has been raised many times compared to the last few years [207]. Nevertheless, the vast growth of IoT is associated with many security and privacy issues (e.g., linkability and tracking). This is because IoT lacks standardized security and privacy guidelines with which such concerns can be addressed. To contribute to such objective, a few research efforts have been conducted. In [15], IoTSE has proposed an IoT framework which provides IoT manufacturers with a comprehensive set of security guidelines as a check-list to simplify the compliance with its framework. In [207], the authors have proposed an IoT security framework known as IoTSEFW to ease the process of integrating security guidelines into IoT industry. More importantly, their security framework will also help companies to accomplish scalability, sustainability, and privacy in their IoT networks. On a government level, UK Government's Department of Digital, Culture, Media and Sport (DCMS), in incorporation with the National Cyber Security Centre (NCSC), in [208] has proposed a set of guidelines. The main goal of such guidelines is to be used by IoT stakeholders to ensure that IoT products are developed with security in mind. DCMS, however, does not provide a comprehensive list of guidelines, nor does it provide compulsory regulation for IoT. Another example is the General Data Protection Regulation (GDPR), which considers the world's powerful data protection rules composed of 99 thorough articles under 11 chapters, and it covers the whole of Europe. That said, GDPR was developed to protect only the personal data of individuals and came into force on 25 May 2018 [208].

7.2. Conclusions

IoT objects are tightly coupled with human beings, since they are involved in too many systems around us such as cars, homes, and hospitals to provide infinite services and solutions. Nevertheless, all those services may encounter enormous risks of security concerns and privacy losses. Therefore, in this survey we perform an in-depth analysis on the first two levels of CISCO'S reference model, namely edge nodes and communication, to mitigate risks associated with these levels. To this end, we first identify IoT security requirements as well as IoT stakeholders. Then, we suggest a comprehensive list of security and privacy guidelines for each previously-mentioned level. We also state those stakeholders who will benefit most from these guidelines to develop secure IoT objects from the start. Furthermore, we recognize a set of proper countermeasures at each level which can be used to implement its proposed guidelines. In this regard, we also illustrate the use of two new emerging technologies called blockchain and SDN to address some IoT security and privacy issues, and we express their limitations in the context of IoT. Moreover, we investigate all possible attacks and threats at each level and state their violated security goals like confidentiality, integrity, and privacy. Furthermore, we briefly discuss the challenges of IoT security and privacy guidelines as well as digital rights management in IoT. Finally, we suggest some open challenges that need further investigation.

Author Contributions: Conceptualization, H.A.A.-G. and D.C.; Methodology, H.A.A.-G.; Validation, D.C. and H.A.A.-G.; Formal Analysis, H.A.A.-G.; Investigation, H.A.A.-G. and D.C.; Resources, H.A.A.-G. and D.C.; Writing—Original Draft Preparation, H.A.A.-G.; Writing—Review & Editing, H.A.A.-G.; Visualization, H.A.A.-G.; Supervision, D.C.; Funding Acquisition, H.A.A.-G. and D.C.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mendez, D.; Papapanagiotou, I.; Yang, B. Internet of Things: Survey on Security and Privacy. *arXiv* **2017**, arXiv:1707.01879.
2. Cirani, S.; Ferrari, G.; Veltri, L. Enforcing Security Mechanisms in the IP-Based Internet of Things: An Algorithmic Overview. *Algorithms* **2013**, *6*, 197–226. [[CrossRef](#)]
3. Riahi Sfar, A.; Natalizio, E.; Challal, Y.; Chtourou, Z. A roadmap for security challenges in the Internet of Things. *Digit. Commun. Netw.* **2018**, *4*, 118–137. [[CrossRef](#)]
4. Kim, D.; Choi, J.Y.; Hong, J.E. Evaluating energy efficiency of Internet of Things software architecture based on reusable software components. *Int. J. Distrib. Sens. Netw.* **2017**, *13*. [[CrossRef](#)]
5. Hong, S.; Kim, D.; Ha, M.; Bae, S.; Park, S.; Jung, W.; Kim, J.E. SNAIL: An IP-based wireless sensor network approach to the internet of things. *IEEE Wirel. Commun.* **2010**, *17*, 34–42. [[CrossRef](#)]
6. Fouladgar, S.; Mainaud, B.; Masmoudi, K.; Afifi, H. *Tiny 3-TLS: A Trust Delegation Protocol for Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2006.
7. Granjal, J.; Monteiro, E.; Silva, J. End-to-end transport-layer security for Internet-integrated sensing applications with mutual and delegated ECC public-key authentication. In Proceedings of the 2013 IFIP Networking Conference, Brooklyn, NY, USA, 22–24 May 2013; pp. 1–9.
8. Brachmann, M.; Keoh, S.L.; Morchon, O.G.; Kumar, S.S. End-to-end transport security in the IP-based internet of things. In Proceedings of the 2012 21st International Conference on Computer Communications and Networks (ICCCN 2012), Munich, Germany, 30 July–2 August 2012; pp. 1–5.
9. Kothmayr, T.; Schmitt, C.; Hu, W.; Brünig, M.; Carle, G. DTLS based security and two-way authentication for the Internet of Things. *Ad Hoc Netw.* **2013**, *11*, 2710–2723. [[CrossRef](#)]
10. Russell, B.; Garlati, C.; Lingenfelter, D. *Security Guidance for Early Adopters of the Internet of Things (IoT)*; Mobile Working Group Peer Reviewed Document; Cloud Security Alliance Publishing: San Francisco, CA, USA, 2015.
11. Lau, D. Secure Bootloader Implementation. 2012. Available online: <https://www.nxp.com/docs/en/application-note/AN4605.pdf> (accessed on 8 April 2019).
12. BITAG. *Internet of Things (IoT) Security and Privacy Recommendations*; BITAG: Denver, CO, USA, 2016.
13. Perera, C.; McCormick, C.; Bandara, A.K.; Price, B.A.; Nuseibeh, B. Privacy-by-Design Framework for Assessing Internet of Things Applications and Platforms. In Proceedings of the 6th International Conference on the Internet of Things—IoT'16, Stuttgart, Germany, 7–9 November 2016; pp. 83–92.
14. U.S. Department of Homeland Security. *Strategic Principles for Securing the Internet of Things (IoT) Introduction and Overview*; U.S. Department of Homeland Security: Washington, DC, USA, 2016; pp. 1–17.
15. IoT Security Foundation. *IoT Security Compliance Framework*; IoT Security Foundation: West Lothian, Scotland, 2016.
16. OWASP. IoT Security Guidance—OWASP. Available online: https://www.owasp.org/index.php/Main_Page (accessed on 8 April 2019).
17. Ross, M.; Jara, A.J.; Cosenza, A. *Baseline Security Recommendations for IoT*; ENISA: Heraklion, Greece, 2017.
18. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [[CrossRef](#)]
19. Atzori, L.; Iera, A.; Morabito, G. The Internet of Things: A survey. *Comput. Netw.* **2010**, *54*, 2787–2805. [[CrossRef](#)]
20. Akram Abdul-Ghani, H.; Konstantas, D.; Mahyoub, M. A Comprehensive IoT Attacks Survey Based on a Building-Blocked Reference Model. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2018**, *9*. [[CrossRef](#)]
21. Cisco. The Internet of Things Reference Model. In *Internet of Things World Forum*; Cisco: San Jose, CA, USA, 2014; pp. 1–12.
22. Zhang, M.; Raghunathan, A.; Jha, N.K. Trustworthiness of medical devices and body area networks. *Proc. IEEE* **2014**, *102*, 1174–1188. [[CrossRef](#)]
23. Li, C.; Raghunathan, A.; Jha, N.K. Hijacking an insulin pump: Security attacks and defenses for a diabetes therapy system. In Proceedings of the 2011 IEEE 13th International Conference on e-Health Networking, Applications and Services, HEALTHCOM 2011, Columbia, MO, USA, 13–15 June 2011; pp. 150–156.

24. Cherdantseva, Y.; Hilton, J. A Reference Model of Information Assurance & Security. In Proceedings of the 2013 International Conference on Availability, Reliability and Security, Regensburg, Germany, 2–6 September 2013.
25. Parno, B.; Perrig, A.; Gligor, V. Distributed Detection of Node Replication Attacks in Sensor Networks. In Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA, USA, 8–11 May 2005.
26. Guzman, A. *IoT Penetration Testing Cookbook*; Packt Publishing: Birmingham, UK, 2017.
27. Das, A.; Rolt, J.D.; Ghosh, S. To cite this version: Secure JTAG implementation using Schnorr Protocol. *J. Electron. Test.* **2013**, *29*, 193–209. [[CrossRef](#)]
28. Vishwakarma, G.; Lee, W. Exploiting JTAG and Its Mitigation in IOT: A Survey. *Future Internet* **2018**, *10*, 121. [[CrossRef](#)]
29. UL LLC. *List of IOT Security Top 20 Design Principles*; White Paper; UL LLC: Northbrook, IL, USA, 2017.
30. Angrishi, K. Turning Internet of Things(IoT) into Internet of Vulnerabilities (IoV): IoT Botnets. *arXiv* **2017**, arXiv:1702.03681.
31. Kanuparthi, A.; Karri, R.; Addepalli, S. Hardware and embedded security in the context of internet of things. In Proceedings of the 2013 ACM workshop on Security, privacy & dependability for cyber vehicles—CyCAR '13, Berlin, Germany, 4 November 2013; pp. 61–64.
32. European Research Cluster on The Internet of Things (IERC). Internet of Things: IoT Governance, Privacy and Security Issues. In *European Research Cluster on the Internet of Things*; IERC: Cork, Ireland, 2015; p. 128.
33. Wahab Ahmed, A.; Muhammad Ahmed, M.; Ahmad Khan, O.; Ali Shah, M. A Comprehensive Analysis on the Security Threats and their Countermeasures of IoT. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* **2017**, *8*, 489–501. [[CrossRef](#)]
34. Baashirah, R.; Abuzneid, A. Survey on prominent RFID authentication protocols for passive tags. *Sensors (Switzerland)* **2018**, *18*, 3584. [[CrossRef](#)]
35. Mohsen Nia, A.; Jha, N.K. A Comprehensive Study of Security of Internet-of-Things. *IEEE Trans. Emerg. Top. Comput.* **2017**, *5*, 586–602. [[CrossRef](#)]
36. Dofe, J.; Frey, J.; Yu, Q. Hardware security assurance in emerging IoT applications. In Proceedings of the IEEE International Symposium on Circuits and Systems, Montreal, QC, Canada, 22–25 May 2016.
37. Abendroth, B.; Kleiner, A.; Nicholas, P. *Cybersecurity Policy for the Internet of Things*; Microsoft Corporation: Redmond, WA, USA, 2017.
38. James, M. *Secure by Design: Improving the Cyber Security of Consumer Internet of Things Report*; Department for Digital, Culture Media & Sport: London, UK, 2017.
39. Alliance, A.S.C. *Embedded Hardware Security for IoT Applications*; Smart Card Alliance: Princeton Junction, NJ, USA, 2016.
40. Corser, G.; Fink, G.A.; Bielby, J. *Internet of Things (IoT) Security Best Practices*; IEEE Internet Technology Policy Community; White Paper; IEEE: Piscataway, NJ, USA, 2017.
41. Microsoft. *The Right Secure Hardware for Your IoT Deployment*; Microsoft Corporation: Redmond, WA, USA, 2017.
42. Cisco. *The Internet of Things: Reduce Security Risks with Automated Policies*; Cisco White Paper; Cisco: San Jose, CA, USA, 2015.
43. IoT Alliance Australia. *Internet of Things Security Guideline*; IoT Alliance Australia: Sydney, Australia, 2017.
44. Syamsuddin, I.; Dillon, T.; Chang, E.; Han, S. A survey of RFID authentication protocols based on Hash-chain method. In Proceedings of the 3rd International Conference on Convergence and Hybrid Information Technology, ICCIT 2008, Busan, Korea, 11–13 November 2008; pp. 559–564. [[CrossRef](#)]
45. Juels, A. RFID Security and Privacy: A Research Survey. *IEEE J. Sel. Areas Commun.* **2006**, *24*, 381–394. [[CrossRef](#)]
46. Rieback, M.R.; Crispo, B.; Tanenbaum, A.S. RFID Guardian: A Battery-Powered Mobile Device for RFID Privacy Management. In *Australasian Conference on Information Security and Privacy*; Springer: Berlin/Heidelberg, Germany, 2005.
47. Ohkubo, M.; Suzuki, K.; Kinoshita, S. Hash-chain based forward- secure privacy protection scheme for low-cost RFID. In Proceedings of the Scandinavian Conference on Information Systems, Odder, Denmark, 5–8 August 2018.

48. Peris-Lopez, P.; Hernandez-Castro, J.C.; Estevez-Tapiador, J.M.; Ribagorda, A. RFID Systems: A Survey on Security Threats and Proposed Solutions. In Proceedings of the IFIP International Conference on Personal Wireless Communications, Albacete, Spain, 20–22 September 2006.
49. Weis, S.A.; Sarma, S.E.; Rivest, R.L.; Engels, D.W. Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. *Secur. Pervasive Comput.* **2003**, *2802*, 201–212.
50. Ye, X.; Feng, J.; Gong, H.; He, C.; Feng, W. An anti-trojans design approach based on activation probability analysis. In Proceedings of the 2015 IEEE International Conference on Electron Devices and Solid-State Circuits, EDSSC 2015, Singapore, 1–4 June 2015; pp. 443–446.
51. Tehranipoor, M.; Koushanfar, F. A survey of hardware trojan taxonomy and detection. *IEEE Des. Test Comput.* **2010**, *27*, 10–25. [[CrossRef](#)]
52. Stajano, F.; Anderson, R. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *International Workshop on Security Protocols*; Christianson, B., Crispo, B., Malcolm, J.A., Roe, M., Eds.; Springer: Berlin/Heidelberg, Germany, 1999; pp. 172–182.
53. Martin, T.; Hsiao, M.; Dong, H.; Krishnaswami, J. Denial-of-service attacks on battery-powered mobile computers. In Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications, Washington, DC, USA, 14–17 March 2004; pp. 309–318.
54. Brandt, A.; Buron, J.; Porcu, G. *Home Automation Routing Requirements in Low-Power and Lossy Networks*; IETF: Fremont, CA, USA, 2010.
55. Harley, D.; Malcho, J. Stuxnet Under the Microscope. Available online: https://www.esetnod32.ru/company/viruslab/analytics/doc/Stuxnet_Under_the_Microscope.pdf (accessed on 8 April 2019).
56. Hernandez, G.; Arias, O.; Buentello, D.; Jin, Y. Smart Nest Thermostat: A Smart Spy in Your Home. *Black Hat USA* **2014**, 1–8.
57. TEMPEST. SideChannel. Available online: <https://sidechannel.tempestsi.com/> (accessed on 8 April 2019).
58. Vuagnoux, M.; Pasini, S. Compromising Electromagnetic Emanations of Wired and Wireless Keyboards. In Proceedings of the 18th Conference on USENIX Security Symposium, Montreal, QC, Canada, 10–14 August 2009.
59. Mohsen Nia, A.; Sur-Kolay, S.; Raghunathan, A.; Jha, N.K. Physiological Information Leakage: A New Frontier in Health Information Security. *IEEE Trans. Emerg. Top. Comput.* **2016**, *4*, 321–334. [[CrossRef](#)]
60. Federal Trade Commission. *Internet of Things: Privacy and Security in a Connected World*; FTC Staff Report; FTC: Washington, DC, USA, 2015.
61. Hancke, G.P. Eavesdropping Attacks on High-Frequency RFID Tokens. Available online: <http://www.rfidblog.org.uk/Hancke-RFIDSec2008-Talk.pdf>; (accessed on 1 April 2019).
62. Zhen-hua, D.; Jin-tao, L.I.; Bo, F.; Zhen-hua, D.; Bo, F. A Taxonomy Model of RFID Security Threats. In Proceedings of the 2008 11th IEEE International Conference on Communication Technology, Hangzhou, China, 10–12 November 2008; pp. 1–4.
63. Juels, A.; Rivest, R.L.; Szydlo, M. The Blocker Tag: Selective Blocking of RFID Tags for The Blocker Tag: Selective Blocking of RFID Tags for Consumer Privacy. In Proceedings of the 10th ACM Conference on Computer and Communication Security—CCS '03, Washington, DC, USA, 27–30 October 2003; p. 103.
64. Iwase, T.; Nozaki, Y.; Yoshikawa, M.; Kumaki, T. Detection technique for hardware Trojans using machine learning in frequency domain. In Proceedings of the 2015 IEEE 4th Global Conference on Consumer Electronics (GCCE), Osaka, Japan, 27–30 October 2015; pp. 185–186.
65. Rooney, C.; Seeam, A.; Bellekens, X. Creation and Detection of Hardware Trojans Using Non-Invasive Off-The-Shelf Technologies. *Electronics* **2018**, *7*, 124. [[CrossRef](#)]
66. Nejat, A.; Mohammdd Hossein Shekarian, S.; Saheb Zamani, M. A study on the efficiency of hardware Trojan detection based on path-delay fingerprinting. *Microprocess. Microsyst.* **2014**, *38*, 246–252. [[CrossRef](#)]
67. Yoshimizu, N. Hardware trojan detection by symmetry breaking in path delays. In Proceedings of the 2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Arlington, VA, USA, 6–7 May 2014; pp. 107–111.
68. Hu, K.; Nowroz, A.N.; Reda, S.; Koushanfar, F. High-Sensitivity Hardware Trojan Detection Using Multimodal Characterization. In Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 18–22 March 2013; pp. 1271–1276.
69. Li, H.; Liu, Q.; Zhang, J. A survey of hardware Trojan threat and defense. *Integration* **2016**, *55*, 426–437. [[CrossRef](#)]

70. Venugopalan, V.; Patterson, C.D. Surveying the Hardware Trojan Threat Landscape for the Internet-of-Things. *J. Hardw. Syst. Secur.* **2018**, *2*, 131–141. [CrossRef]
71. Msgna, M.; Markantonakis, K.; Naccache, D.; Mayes, K. *Verifying Software Integrity in Embedded Systems: A Side Channel Approach*; Springer: Cham, Switzerland, 2014; pp. 261–280.
72. Msgna, M.; Markantonakis, K.; Mayes, K. The B-Side of Side Channel Leakage: Control Flow Security in Embedded Systems. In Proceedings of the International Conference on Security and Privacy in Communication Systems, Sydney, Australia, 25–27 September 2013; pp. 288–304.
73. Stergiou, P.; Maniatakis, M.; Konstantinou, C.; Robison, P.; Lee, S.; Kim, S.; Wang, X.; Karri, R. Malicious Firmware Detection with Hardware Performance Counters. *IEEE Trans. Multi-Scale Comput. Syst.* **2016**, *2*, 160–173. [CrossRef]
74. Rosenfeld, K.; Karri, R. Attacks and defenses for JTAG. *IEEE Des. Test Comput.* **2010**, *27*, 36–47. [CrossRef]
75. Clark, C.J. Anti-tamper JTAG TAP design enables DRM to JTAG registers and P1687 on-chip instruments. In Proceedings of the 2010 IEEE International Symposium on Hardware-Oriented Security and Trust, HOST 2010, Anaheim, CA, USA, 13–14 June 2010; pp. 19–24.
76. Backer, J.; Hely, D.; Karri, R. Secure and Flexible Trace-Based Debugging of Systems-on-Chip. *ACM Trans. Des. Autom. Electron. Syst.* **2017**, *22*, 1–25. [CrossRef]
77. Peris-lopez, P.; Hernandez-castro, J.C.; Estevez-tapiador, J.M.; Ribagorda, A. M2AP: A Minimalist Mutual-Authentication Protocol for Low-cost RFID Tags Pedro. In Proceedings of the Third International Conference, UIC 2006, Wuhan, China, 3–6 September 2006.
78. Jung, M.; Fiedler, H.L.; Fiedler, H.L.; Lerch, R.G. 8-bit-microcontroller system with area efficient AES coprocessor for transponder applications. In Proceedings of the Workshop on RFID and Lightweight Crypto, Graz, Austria, 14–15 July 2005.
79. Labbi, Z.; Senhadji, M.; Maarof, A.; Belkasmi, M. Symmetric Encryption Algorithm for RFID Systems Using a Dynamic Generation of Key. *Int. J. Comput. Sci. Issues* **2018**, *15*, 25–33. [CrossRef]
80. Choi, E.Y.; Lee, S.M.; Lee, D.H. Efficient RFID Authentication Protocol for Ubiquitous Computing Environment. In Proceedings of the International Conference on Embedded and Ubiquitous Computing, Nagasaki, Japan, 6–9 December 2005; pp. 945–954.
81. Dimitriou, T. A Lightweight RFID Protocol to Protect against Traceability and Cloning Attacks. In Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05), Athens, Greece, 5–9 September 2005.
82. Lee, S.M.; Hwang, Y.J.; Lee, D.H.; Lim, J.I. *Efficient Authentication for Low-Cost RFID Systems*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 619–627.
83. Peris-Lopez, P.; Hernandez-Castro, J.C.; Estevez-Tapiador, J.M.; Ribagorda, A. AAP: A Minimalist 1286 Mutual-Authentication Protocol for Low-cost RFID Tags. In Proceedings of the International Conference on Ubiquitous Intelligence and Computing, Orange County, CA, USA, 17–21 September 2006.
84. Molnar, D.; Wagner, D. Privacy and security in library RFID. In Proceedings of the 11th ACM Conference on Computer and Communications Security—CCS '04, Washington, DC, USA, 25–29 October 2004; p. 210.
85. Surendran, S.; Nassef, A.; Beheshti, B.D. A survey of cryptographic algorithms for IoT devices. In Proceedings of the IEEE Long Island Systems, Applications and Technology Conference, LISAT, New York, NY, USA, 4 May 2018; pp. 1–8.
86. Carluccio, D.; Lemke, K.; Paar, C. Electromagnetic Side Channel Analysis of a Contactless Smart Card: First Results. In Proceedings of the ECRYPT Workshop on RFID and Lightweight Crypto, Graz, Austria, 14–15 July 2005.
87. Zhang, M.; Jha, N.K. 0 FinFET-Based Power Management for Improved DPA Resistance with Low Overhead. *ACM J. Emerg. Technol. Comput. Syst.* **2011**, *7*, 10. [CrossRef]
88. Osvik, D.A.; Shamir, A.; Tromer, E. Cache Attacks and Countermeasures: The Case of AES. In Proceedings of the Cryptographers' Track at the RSA Conference, San Jose, CA, USA, 13–17 February 2006.
89. Sen, J. Security in Wireless Sensor Networks. CoRR, abs/1301.5065. 2013. Available online: <https://arxiv.org/ftp/arxiv/papers/1301/1301.5065.pdf> (accessed on 8 April 2019).
90. Rosenfeld, K.; Gavas, E.; Karri, R. Sensor physical unclonable functions. In Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust, Anaheim, CA, USA, 13–14 June 2010; pp. 112–117.

91. Guin, U.; Dimase, D.; Tehranipoor, M. *Counterfeit Integrated Circuits: Detection, Avoidance, and the Challenges Ahead*; Springer: Berlin/Heidelberg, Germany, 2014.
92. Cortese, P.F.; Gemmiti, F.; Palazzi, B.; Pizzonia, M.; Rimondini, M. Efficient and practical authentication of PUF-based RFID tags in supply chains. In Proceedings of the 2010 IEEE International Conference on RFID-Technology and Applications, Guangzhou, China, 17–19 June 2011; pp. 182–188.
93. Moriyama, D.; Matsuo, I.; Yung, M. PUF-Based RFID Authentication Secure and Private under Memory Leakage. *IACR Cryptol. ePrint Arch.* **2013**, *3*, 61–83.
94. Hristozov, S.; Heyszl, J.; Wagner, S.; Sigl, G. Practical Runtime Attestation for Tiny IoT Devices. In Proceedings of the 2018 Workshop on Decentralized IoT Security and Standards, San Diego, CA, USA, 18 February 2018.
95. Trusted Computing Group. *TPM Main Specification*; Trusted Computing Group: Beaverton, OR, USA, 2011.
96. Limited, A. ARM Security Technology Building a Secure System using TrustZone @Technology. Available online: <http://infocenter.arm.com/help/topic/com.arm.doc.pr29-genc-009492c/PRD29-GENC-009492C-trustzone-security-whitepaper.pdf> (accessed on 1 April 2019).
97. Law, Y.; Zhang, Y.; Jin, J.; Palaniswami, M.; Havinga, P. Secure Rateless Deluge: Pollution-Resistant Reprogramming and Data Dissemination for Wireless Sensor Networks. *EURASIP J. Wirel. Commun. Network.* **2011**, *2011*, 685219. [CrossRef]
98. Saiful Islam Mamun, M.; Sultanul Kabir, A.; Sakhawat Hossen, M.; Hayat Khan, M. Policy Based Intrusion Detection and Response System in Hierarchical WSN Architecture. *arXiv* **2012**, arXiv:1209.1678.
99. Zhijie, H.; Ruchuang, W. 2012 International Conference on Solid State Devices and Materials Science Intrusion Detection for Wireless Sensor Network Based on Traffic Prediction Model peer-review under responsibility of [name organizer]. *Phys. Procedia* **2012**, *25*, 2072–2080. [CrossRef]
100. Hindy, H.; Brosset, D.; Bayne, E.; Seeam, A.; Tachtatzis, C.; Atkinson, R.; Bellekens, X. A Taxonomy and Survey of Intrusion Detection System Design Techniques, Network Threats and Datasets. *arXiv* **2018**, arXiv:1806.03517.
101. Mitchell, R.; Chen, I.R. A survey of intrusion detection techniques for cyber-physical systems. *ACM Comput. Surv.* **2014**, *46*, 1–29. [CrossRef]
102. Butun, I.; Morgera, S.D.; Sankar, R. 002—A Survey of Intrusion Detection Systems in Wireless Sensor Networks. *IEEE Sens. J.* **2014**, *14*, 1370–1379. [CrossRef]
103. Mozzaquatro, B.A.; Agostinho, C.; Goncalves, D.; Martins, J.; Jardim-goncalves, R. An Ontology-Based Cybersecurity Framework for the Internet of Things. *Sensors* **2018**, *18*, 3053. [CrossRef] [PubMed]
104. INTEL COMPANY. Managing the IoT Lifecycle from Design through End-of-Life. Available online: <https://www.iotone.com/guide/managing-the-iot-lifecycle-from-design-through-end-of-life/g923> (accessed on 1 April 2019).
105. Samyde, J.J.Q. ElectroMagnetic Analysis (EMA): Measures and Counter-measures for Smart Card. In Proceedings of the International Conference on Research in Smart Cards (E-smart 2001), Cannes, France, 19–21 September 2001.
106. Juels, A.; Brainard, J. Soft Blocking: Flexible Blocker Tags on the Cheap. In Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society, Washington, DC, USA, 28 October 2004.
107. Chen, Y.Y.; Lu, J.C.; Chen, S.I.; Jan, J.K. A low-cost RFID authentication protocol with location privacy protection. In Proceedings of the 5th International Conference on Information Assurance and Security, IAS 2009, Xi'an, China, 18–20 August 2009; pp. 109–113.
108. Lesperance, N.; Kulkarni, S.; Cheng, K.T. Hardware Trojan Detection Using Exhaustive Testing of k-bit Subspaces. In Proceedings of the The 20th Asia and South Pacific Design Automation Conference, Chiba, Japan, 19–22 January 2015; pp. 755–760.
109. Subhra Chakraborty, R.; Wolff, F.; Paul, S.; Papachristou, C.; Bhunia, S. *MERO: A Statistical Approach for Hardware Trojan Detection*; Technical Report; Springer: Berlin/Heidelberg, Germany, 2009.
110. Pierce, L.; Tragoudas, S. Multi-level secure JTAG architecture. In Proceedings of the 2011 IEEE 17th International On-Line Testing Symposium, IOLTS, Athens, Greece, 13–15 July 2011; pp. 208–209.
111. Mauw, S.; Piramuthu, S. A PUF-based authentication protocol to address ticket-switching of RFID-tagged items. In Proceedings of the International Workshop on Security and Trust Management (STM 2012), Pisa, Italy, 13–14 September 2012; pp. 209–224.

112. Dragomir, D.; Gheorghe, L.; Costea, S.; Radovici, A. A Survey on Secure Communication Protocols for IoT Systems. In Proceedings of the 2016 International Workshop on Secure Internet of Things (SIoT), Crete, Greece, 26–30 September 2016.
113. Lin, J.; Yu, W.; Zhang, N.; Yang, X.; Zhang, H.; Zhao, W. A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. *IEEE Internet Things J.* **2017**, *4*, 1125–1142. [[CrossRef](#)]
114. Deshmukh, S.; Sonavane, S.S. Security protocols for Internet of Things: A survey. In Proceedings of the 2017 International Conference On Nextgen Electronic Technologies: Silicon to Software, ICNETS2 2017, Chennai, India, 23–25 March 2017; pp. 71–74.
115. Agrawal, S. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *Abakos* **2015**, *1*, 78–95. [[CrossRef](#)]
116. Venčkauskas, A.; Morkevicius, N.; Bagdonas, K.; Damaševičius, R.; Maskeliūnas, R. A lightweight protocol for secure video streaming. *Sensors* **2018**, *18*, 1554. [[CrossRef](#)] [[PubMed](#)]
117. Tomić, I.; McCann, J.A. A Survey of Potential Security Issues in Existing Wireless Sensor Network Protocols. *IEEE Internet Things J.* **2017**, *4*, 1910–1923. [[CrossRef](#)]
118. Jebri, S.; Abid, M.; Bouallegue, A. An efficient scheme for anonymous communication in IoT. In Proceedings of the 2015 11th International Conference on Information Assurance and Security (IAS), Marrakech, Morocco, 14–16 December 2015; pp. 7–12.
119. Barki, A.; Bouabdallah, A.; Gharout, S.; Traore, J. M2M Security: Challenges and Solutions. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1241–1254. [[CrossRef](#)]
120. Liu, W.; Zhou, G.; Wei, J.; Hu, X.; Kumari, S. Security enhanced and cost-effective user authentication scheme for wireless sensor networks. *Inf. Technol. Control* **2018**, *47*, 275–294. [[CrossRef](#)]
121. Mayzaud, A.; Badonnel, R.; Chrisment, I. A taxonomy of attacks in RPL-based internet of things. *Int. J. Netw. Sec.* **2016**, *18*, 459–473.
122. Sheng, Z.; Yang, S.; Yu, Y.; Vasilakos, A.; Mccann, J.; Leung, K. A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities. *IEEE Wirel. Commun.* **2013**, *20*, 91–98. [[CrossRef](#)]
123. Sarikaya, B.; Ohba, Y.; Moskowitz, R.; Cao, Z.; Cragie, R. *Security Bootstrapping Solution for Resource-Constrained Devices*; Technical Report for the Internet Engineering Task Force; IETF: Fremont, CA, USA, 22 June 2012.
124. Heer, T.; Garcia-Morchon, O.; Hummen, R.; Keoh, S.L.; Kumar, S.S.; Wehrle, K. Security Challenges in the IP-based Internet of Things. *Wirel. Person. Commun.* **2011**, *61*, 527–542. [[CrossRef](#)]
125. Vasilomanolakis, E.; Daubert, J.; Luthra, M.; Gazis, V.; Wiesmaier, A.; Kikiras, P. On the Security and Privacy of Internet of Things Architectures and Systems. In Proceedings of the 2015 International Workshop on Secure Internet of Things, SIoT 2015, Vienna, Austria, 21–25 September 2015; pp. 49–57.
126. Han, J.; Kamber, M.; Pei, J. *Data Mining. Concepts and Techniques*, 3rd ed.; The Morgan Kaufmann Series in Data Management Systems; Morgan Kaufmann: Burlington, MA, USA, 2011.
127. Miorandi, D.; Sicari, S.; De Pellegrini, F.; Chlamtac, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [[CrossRef](#)]
128. Alharby, S.; Harris, N.; Weddell, A.; Reeve, J. Impact of duty cycle protocols on security cost of IoT. In Proceedings of the 2018 9th International Conference on Information and Communication Systems, ICICS 2018, Irbid, Jordan, 3–5 April 2018; pp. 25–30.
129. Domingo-Pascual, J.J.; Shavitt, Y.; Uhlig, S. Traffic Monitoring and Analysis. In Proceedings of the Third International Workshop, TMA 2011, Vienna, Austria, 27 April 2011; p. 196.
130. Naoui, S.; Elhdhili, M.E.; Saidane, L.A. Security analysis of existing IoT key management protocols. In Proceedings of the IEEE/ACS International Conference on Computer Systems and Applications, AICCSA, Hammamet, Tunisia, 30 October–3 November 2017; pp. 1–7.
131. Bormann, C.; Castellani, A.P.; Shelby, Z. CoAP: An Application Protocol for Billions of Tiny Internet Nodes. *IEEE Internet Comput.* **2012**, *16*, 62–67. [[CrossRef](#)]
132. Raza, S.; Tralbalza, D.; Voigt, T. 6LoWPAN compressed DTLS for CoAP. In Proceedings of the IEEE International Conference on Distributed Computing in Sensor Systems, DCOSS, Hangzhou, China, 16–18 May 2012; pp. 287–289.
133. Singh, M.; Rajan, M.A.; Shivraj, V.L.; Balamuralidhar, P. Secure MQTT for Internet of Things (IoT). In Proceedings of the 2015 5th International Conference on Communication Systems and Network Technologies, CSNT, Gwalior, India, 4–6 April 2015; pp. 746–751.

134. Granjal, J.; Monteiro, E.; Sa Silva, J. Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 1294–1312. [[CrossRef](#)]
135. Keoh, S.L.; Kumar, S.S.; Garcia-Morchon, O.; Dijk, E. *DTLS-Based Mul-Ticast Security for Low-Power and Lossy*; Technical Report for the Internet Engineering Task Force; IETF: Fremont, CA, USA, 2015; pp. 1–22.
136. Sethi, M.; Arkko, J.; Keranen, A. End-to-end security for sleepy smart object networks. In Proceedings of the Conference on Local Computer Networks, LCN, Clearwater, FL, USA, 22–25 October 2012; pp. 964–972.
137. Misra, S.; Vaish, A. Reputation-based role assignment for role-based access control in wireless sensor networks. *Comput. Commun.* **2011**, *34*, 281–294. [[CrossRef](#)]
138. Walters, J.P.; Liang, Z.; Shi, W.; Chaudhary, V. Wireless Sensor Network Security: A Survey. In *Security in Distributed, Grid, Mobile, and Pervasive Computing*; CRC Press: Boca Raton, FL, USA, 2007.
139. Rajan, A.; Jithish, J.; Sankaran, S. Sybil attack in IOT: Modelling and defenses. In Proceedings of the 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 13–16 September 2017.
140. Shanmuganathan, V.; Anand, M.T. A Survey on Gray Hole Attack in MANET. *IRACST-Int. J. Comput. Netw. Wirel. Commun. (IJCNWC)* **2012**, *2*, 647–650.
141. Kaushal Shahpur, K.; Sahni, V. DoS Attacks on different Layers of WSN: A Review. *Int. J. Comput. Appl.* **2015**, *130*, 8–11. [[CrossRef](#)]
142. Phelan, T. Datagram Transport Layer Security (DTLS) over the Datagram Congestion Control Protocol (DCCP). 2008. Available online: <https://tools.ietf.org/html/draft-ietf-dccp-dtls-06> (accessed on 8 April 2019).
143. Moskowitz, R.; Nikander, P.; Jokela, T.H. *Host Identity Protocol*; Technical Report for Internet Engineering Task Force; IETF: Fremont, CA, USA, 2008.
144. Kaufman, C. *Internet Key Exchange (IKEv2) Protocol*; Technical Report; Internet Engineering Task Force (IETF): Fremont, CA, USA, 2005.
145. Moskowitz, R. HIP Diet EXchange (DEX). 2011. Available online: <https://tools.ietf.org/html/draft-moskowitz-hip-rg-dex-05> (accessed on 8 April 2019).
146. Wook Jung, S.; Jung, S. Secure Bootstrapping and Rebootstrapping for Resource-Constrained Thing in Internet of Things. *Int. J. Distrib. Sens. Netw.* **2015**. [[CrossRef](#)]
147. Sarikaya, B.; Sethi, M.; Sangi, A.R. *Secure IoT Bootstrapping: A Survey*; Technical Report for Internet Engineering Task Force; IETF: Fremont, CA, USA, 2018.
148. Montenegro, G.; Kushalnagar, N.; Hui, J.; Culler, D. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*; Technical Report for Internet Engineering Task Force; IETF: Fremont, CA, USA, 2007.
149. Watteyne, T.; Palattella, M.; Grieco, L. Using IEEE 802.15.4e Time-Slotted Channel Hopping (TSCH) in the Internet of Things (IoT): Problem Statement. 2015. Available online: <https://www.rfc-editor.org/rfc/rfc7554.txt> (accessed on 8 April 2019).
150. Roman, R.; Alcaraz, C.; Lopez, J.; Sklavos, N. Key Management Systems for Sensor Networks in the Context of the Internet of Things. *Comput. Electr. Eng.* **2011**, *37*, 147–159. [[CrossRef](#)]
151. ArchRock Corporation. Phynet n4x Series. 2008. Available online: <https://www.businesswire.com/news/home/20081014005655/en/Arch-Rock-Adds> (accessed on 8 April 2019).
152. Moskowitz, R.; Hummen, R. *HIP Diet EXchange (DEX)*; Internet Engineering Task Force (IETF): Fremont, CA, USA, 2017.
153. Dierks, T.; Allen, C. *The TLS Protocol*; Internet Engineering Task Force (IETF): Fremont, CA, USA, 1999.
154. Hummen, R.; Ziegeldorf, J.H.; Shafagh, H.; Raza, S.; Wehrle, K. Towards Viable Certificate-based Authentication for the Internet of Things. In Proceedings of the 2nd ACM Workshop on Hot Topics on Wireless Network Security and Privacy, Budapest, Hungary, 19 April 2013; pp. 37–41.
155. Granjal, J.; Monteiro, E.; Silva, J.S. Network-layer security for the Internet of Things using TinyOS and BLIP. *Int. J. Commun. Syst.* **2014**, *27*, 1938–1963. [[CrossRef](#)]
156. Raza, S.; Voigt, T.; Jutvik, V. Secure communication for the Internet of Things—A comparison of link-layer security and IPsec for 6LoWPAN. *Int. J. Appl. Eng. Res.* **2014**, *9*, 5968–5974. [[CrossRef](#)]
157. Raza, S.; Voigt, T.; Jutvik, V. Lightweight IKEv2: A Key Management Solution for both the Compressed IPsec and the IEEE 802.15.4 Security. In Proceedings of the IETF Workshop on Smart Object Security, Paris, France, 23 March 2012.

158. Hummen, R.; Hiller, J.; Wirtz, H.; Henze, M.; Shafagh, H.; Wehrle, K. 6LoWPAN fragmentation attacks and mitigation mechanisms. In Proceedings of the Sixth ACM conference on Security and Privacy in Wireless and Mobile Networks—WiSec '13, Budapest, Hungary, 17–19 April 2013; p. 55.
159. Raza, S.; Duquennoy, S.; Chung, T.; Yazar, D.; Voigt, T.; Roedig, U. Securing Communication in 6LoWPAN with Compressed IPsec. In Proceedings of the 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), Barcelona, Spain, 27–29 June 2011.
160. Granjal, J.; Monteiro, E.; Silva, J.S. Enabling Network-Layer Security on IPv6 Wireless Sensor Networks. In Proceedings of the 2010 IEEE Global Telecommunications Conference GLOBECOM, Miami, FL, USA, 6–10 December 2010; pp. 1–6.
161. Hummen, R.; Wirtz, H.; Ziegeldorf, J.H.; Hiller, J.; Wehrle, K. Tailoring end-to-end IP security protocols to the internet of things. In Proceedings of the International Conference on Network Protocols, ICNP, Goettingen, Germany, 7–10 October 2013.
162. Winter, T.; Thubert, P.; Brandt, A.; Hui, J.; Kelsey, R.; Levis, P.; Pister, K.; Struik, R.; Vasseur, J.P.; Alexander, R. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*; RFC 6550; Internet Engineering Task Force (IETF): Fremont, CA, USA, 2012.
163. Tsao, T.; Alexander, R.; Dohler, M. *A Security Threat Analysis for Routing Protocol for Low-Power and Lossy Networks (RPL)*; RFC7416; Internet Engineering Task Force (IETF): Fremont, CA, USA, 2014; p. 131.
164. Le, A.; Loo, J.; Lasebae, A.; Vinel, A.; Chen, Y.; Chai, M. The impact of rank attack on network topology of routing protocol for low-power and lossy networks. *IEEE Sens. J.* **2013**, *13*, 3685–3692. [[CrossRef](#)]
165. Dvir, A.; Holczer, T.; Buttyan, L. VeRA—Version number and rank authentication in RPL. In Proceedings of the 8th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS, Valencia, Spain, 17–21 October 2011; pp. 709–714.
166. Weekly, K.; Pister, K. Evaluating sinkhole defense techniques in RPL networks. In Proceedings of the International Conference on Network Protocols, ICNP, Austin, TX, USA, 30 October–2 November 2012; pp. 1–6.
167. Kim, H.S.; Ko, J.G.; Culler, D.E.; Paek, J. Challenging the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL): A Survey. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2502–2525. [[CrossRef](#)]
168. Hartke, K. *Practical Issues with Datagram Transport Layer Security in Constrained Environments*; DICE Working Group: Fremont, CA, USA, 2014.
169. Keoh, S.; Kumar, S.; Shelby, Z. *Profiling of DTLS for CoAP-Based IoT Applications*; Technical Report for Internet Engineering Task Force; IETF: Fremont, CA, USA, 2013.
170. Kothmayr, T.; Schmitt, C.; Hu, W.; Brunig, M.; Carle, G. A DTLS based end-to-end security architecture for the Internet of Things with two-way authentication. In Proceedings of the Conference on Local Computer Networks, LCN, Clearwater, FL, USA, 22–25 October 2012.
171. Granjal, J.; Monteiro, E.; Silva, J.S. Application-layer security for the WoT: Extending CoAP to support end-to-end message security for internet-integrated sensing applications. In Proceedings of the 11th Wired/Wireless Internet Communication, St. Petersburg, Russia, 5–7 June 2013.
172. Ramsdell, B. *Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification*; IETF: Fremont, CA, USA, 2004.
173. Abduvaliyev, A.; Pathan, A.S.K.; Zhou, J.; Roman, R.; Wong, W.C. On the vital areas of intrusion detection systems in wireless sensor networks. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1223–1237. [[CrossRef](#)]
174. da Silva, A.P.R.; Martins, M.H.T.; Rocha, B.P.S.; Loureiro, A.A.F.; Ruiz, L.B.; Wong, H.C. Decentralized intrusion detection in wireless sensor networks. In Proceedings of the 1st ACM International Workshop on Quality of service & Security in Wireless and Mobile Networks—Q2SWinet '05, Patras, Greece, 13 October 2005; p. 16.
175. Raza, S.; Wallgren, L.; Voigt, T. SVELTE: Real-time intrusion detection in the Internet of Things. *Ad Hoc Netw.* **2013**, *11*, 2661–2674. [[CrossRef](#)]
176. Liu, C.; Yang, J.; Chen, R.; Zhang, Y.; Zeng, J. Research on immunity-based intrusion detection technology for the Internet of Things. In Proceedings of the 2011 7th International Conference on Natural Computation, ICNC, Shanghai, China, 26–28 July 2011; pp. 212–216.
177. Gaurav, K.; Goya, P.V.A. IoT transaction security. In Proceedings of the 5th International Conference on the Internet of Things (IoT), Seoul, Korea, 26–28 October 2015.

178. Biswas, K.; Muthukkumarasamy, V. Securing smart cities using blockchain technology. In Proceedings of the 18th IEEE International Conference on High Performance Computing and Communications, 14th IEEE International Conference on Smart City and 2nd IEEE International Conference on Data Science and Systems, HPCC/SmartCity/DSS, Sydney, Australia, 12–14 December 2016; pp. 1392–1393.
179. Kokoris-Kogias, L.; Gasser, L.; Khoffi, I.; Jovanovic, P.; Gailly, N.; Ford, B. Managing Identities Using Blockchains and CoSi. In Proceedings of the 9th Workshop on Hot Topics in Privacy Enhancing Technologies (HotPETs), Darmstadt, Germany, 19–22 July 2016.
180. Bahga, A.; Madiseti, V.K. Blockchain Platform for Industrial Internet of Things. *J. Softw. Eng. Appl.* **2016**, *9*, 533–546. [[CrossRef](#)]
181. Otte, P.; de Vos, M.; Pouwelse, J. TrustChain: A Sybil-resistant scalable blockchain. *Future Gener. Comput. Syst.* **2017**. [[CrossRef](#)]
182. Hashemi, S.H.; Faghri, F.; Rausch, P.; Campbell, R.H. World of empowered IoT users. In Proceedings of the 2016 IEEE 1st International Conference on Internet-of-Things Design and Implementation, IoTDL, Berlin, Germany, 4–8 April 2016; pp. 13–24.
183. Yue, X.; Wang, H.; Jin, D.; Li, M.; Jiang, W. Healthcare Data Gateways: Found Healthcare Intelligence on Blockchain with Novel Privacy Risk Control. *J. Med. Syst.* **2016**, *40*, 218. [[CrossRef](#)]
184. Conoscenti, M.; Carlos De Martin, J. IOT_Blockchain for the Internet of Things: a Systematic Literature Review. In Proceedings of the Third International Symposium on Internet of Things: Systems, Management and Security, Agadir, Morocco, 29 November–2 December 2016; pp. 1–6.
185. Khan, M.A.; Salah, K. IoT security: Review, blockchain solutions, and open challenges. *Future Gener. Comput. Syst.* **2018**, *82*, 395–411. [[CrossRef](#)]
186. Reyna, A.; Martín, C.; Chen, J.; Soler, E.; Díaz, M. On blockchain and its integration with IoT. Challenges and opportunities. *Future Gener. Comput. Syst.* **2018**, *88*, 173–190. [[CrossRef](#)]
187. Kouicem, D.E.; Bouabdallah, A.; Lakhlef, H.; Kouicem, D.E.; Bouabdallah, A.; Lakhlef, H. Internet of things security: A top-down survey. *Comput. Netw.* **2018**, *141*, 199–221. [[CrossRef](#)]
188. Hu, P. A System Architecture for Software-Defined Industrial Internet of Things. In Proceedings of the 2015 IEEE International Conference on Ubiquitous Wireless Broadband, ICUBW, Montreal, QC, Canada, 4–7 October 2015; pp. 1–5. [[CrossRef](#)]
189. Flauzac, O.; Gonzalez, C.; Hachani, A.; Nolot, F. SDN Based Architecture for IoT and Improvement of the Security. In Proceedings of the IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, WAINA, Gwangju, Korea, 25–27 March 2015.
190. Vandana, C. Security improvement in IoT based on Software Defined Networking (SDN). *Int. J. Sci. Eng. Technol. Res. (IJSETR)* **2016**, *5*, 2327–4662.
191. Gonzalez, C.; Charfadine, S.M.; Flauzac, O.; Nolot, F. SDN-based security framework for the IoT in distributed grid. In Proceedings of the International Multidisciplinary Conference on Computer and Energy Science, SpliTech, Split, Croatia, 13–15 July 2016; pp. 1–5.
192. Aragon, S.; Tiloca, M.; Maass, M.; Hollick, M.; Raza, S. ACE of spades in the iot security game: A flexible ipsec security profile for access control. In Proceedings of the 2018 IEEE Conference on Communications and Network Security, CNS, Beijing, China, 30 May–1 June 2018.
193. Bergmann, O.; Bormann, C.; Tzi, U.B.; Ab, E.; Seitz, L. Datagram Transport Layer Security (DTLS) Profile for Authentication and Authorization for Constrained Environments (ACE). 2019, pp. 1–19. Available online: <https://tools.ietf.org/html/draft-ietf-ace-dtls-authorize-03> (accessed on 8 April 2019).
194. Selander, G.; Ab, S.; Tschofenig, H. Authentication and Authorization for Constrained Environments (ACE). 2018, pp. 1–66. Available online: <https://tools.ietf.org/html/draft-ietf-ace-oauth-authz-07> (accessed on 8 April 2019).
195. Ab, E.; Seitz, L. Object Security for Constrained RESTful Environments (OSCORE). 2019. Available online: <https://tools.ietf.org/html/draft-ietf-core-object-security-16> (accessed on 8 April 2019).
196. Kubesch, A.S.; Wicker, S. Digital rights management: The cost to consumers. *Proc. IEEE* **2015**, *103*, 726–733. [[CrossRef](#)]
197. Microsoft Corporation. *Digital Transformation with IoT: How OEMs and ISVs Can Lead the Way*; Microsoft Corporation: Redmond, WA, USA, 2017.
198. Lee, C.C.; Li, C.T.; Chen, Z.W.; Lai, Y.M. A biometric-based authentication and anonymity scheme for digital rights management system. *Inf. Technol. Control* **2018**, *47*, 262–274. [[CrossRef](#)]

199. Chen, C.L.; Chin-Ling. A secure and traceable E-DRM system based on mobile device. *Expert Syst. Appl.* **2008**, *35*, 878–886. [[CrossRef](#)]
200. Chang, C.C.; Yang, J.H.; Wang, D.W. An efficient and reliable E-DRM scheme for mobile environments. *Expert Syst. Appl.* **2010**, *37*, 6176–6181. [[CrossRef](#)]
201. Mishra, D.; Das, A.K.; Mukhopadhyay, S. An anonymous and secure biometric-based enterprise digital rights management system for mobile environment. *Secur. Commun. Netw.* **2015**, *8*, 3383–3404. [[CrossRef](#)]
202. Newman, R.; Doody, P.; Trebar, M.; Okoke, U. Rights management to enable a true Internet of Things. In Proceedings of the IEEE International Conference on Intelligence and Security Informatics: Cybersecurity and Big Data, ISI, Tucson, AZ, USA, 2016; pp. 1–6.
203. Yu, S.; Member, S. Big Privacy: Challenges and Opportunities of Privacy Study in the Age of Big Data. *IEEE Access* **2016**, *4*, 2751–2763. [[CrossRef](#)]
204. Doddapaneni, K.; Lakkundi, R.; Rao, S.; Kulkarni, S.G.; Bhat, B. Secure FoTA Object for IoT. In Proceedings of the 2017 IEEE 42nd Conference on Local Computer Networks Workshops, LCN Workshops 2017, Singapore, 9 October 2017; pp. 154–159.
205. Blake, I.F.; Seroussi, G.G.; Smart, N.P.N.P.; Cassels, J.W.S. *Advances in Elliptic Curve Cryptography*; Cambridge University Press: New York, NY, USA, 2005; p. 281.
206. Kaźmierski, T.J.; Beeby, S. *Energy Harvesting Systems: Principles, Modeling and Applications*; Springer: New York, NY, USA, 2011; p. 163.
207. Saleem, J.; Hammoudeh, M.; Raza, U.; Adebisi, B.; Ande, R. IoT standardisation: Challenges, perspectives and solution. In Proceedings of the 2nd International Conference on Future Networks and Distributed Systems—ICFNDS '18, Amman, Jordan, 26–27 June 2018; ACM Press: New York, NY, USA, 2018; pp. 1–9.
208. DDCMS. *Code of Practice for Consumer IoT Security*; Technical Report; DDCMS: Wakefield, UK, 2018.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).