




Security of Internet of Things for a Reliable Internet of Services

Ahmet Arı̇s¹, Sema F. Oktuđ¹, and Thiemo Voigt²

¹ Faculty of Computer and Informatics Engineering,
Istanbul Technical University, Istanbul, Turkey
{[arisahmet](mailto:arisahmet@itu.edu.tr),[oktug](mailto:oktug@itu.edu.tr)}@itu.edu.tr

² Swedish Institute of Computer Science (SICS), Kista, Sweden
thiemo@sics.se

Abstract. The Internet of Things (IoT) consists of resource-constrained devices (e.g., sensors and actuators) which form low power and lossy networks to connect to the Internet. With billions of devices deployed in various environments, IoT is one of the main building blocks of future Internet of Services (IoS). Limited power, processing, storage and radio dictate extremely efficient usage of these resources to achieve high reliability and availability in IoS. Denial of Service (DoS) and Distributed DoS (DDoS) attacks aim to misuse the resources and cause interruptions, delays, losses and degrade the offered services in IoT. DoS attacks are clearly threats for availability and reliability of IoT, and thus of IoS. For highly reliable and available IoS, such attacks have to be prevented, detected or mitigated autonomously. In this study, we propose a comprehensive investigation of Internet of Things security for reliable Internet of Services. We review the characteristics of IoT environments, cryptography-based security mechanisms and D/DoS attacks targeting IoT networks. In addition to these, we extensively analyze the intrusion detection and mitigation mechanisms proposed for IoT and evaluate them from various points of view. Lastly, we consider and discuss the open issues yet to be researched for more reliable and available IoT and IoS.

Keywords: IoT · IoT security · IoS · DoS · DDoS
Internet of Things · Internet of Services · Reliable IoS

1 Introduction

Internet of Things is a network of sensors, actuators, embedded and wearable devices that can connect to the Internet. Billions of devices are expected to be part of this network and make houses, buildings, cities and many other deployment areas smarter [17]. In order to reach populations as much as billions, elements of IoT network are expected to be cheap and small form-factor devices with limited resources.

IoT is a candidate technology in order to realize the future Internet of Services and Industry 4.0 revolution. Accommodation of billions of devices with sensing

and/or actuation capabilities will introduce crucial problems with management, interoperability, scalability, reliability, availability and security. Autonomous control and reliability of future IoS are directly related to reliability and availability of IoT. However, there are serious threats for IoT, which aim to degrade the performance of the network, deplete the batteries of the devices and cause packet losses and delays. These attacks are called as Denial of Service attacks, which are already notorious for their effects in existing communication systems. Limited power, processing, storage and radio dictate extremely efficient usage of these resources to achieve high reliability and availability in IoS. However, DoS and DDoS attacks aim to misuse the resources and cause interruptions, delays, losses and degrade the offered services in IoT. DoS attacks are clearly threats for availability and reliability of IoT, and thus of IoS. For highly reliable and available IoS, such attacks have to be prevented, detected or mitigated autonomously.

DoS and DDoS attacks can target any communication system and cause devastation. Such attacks make use of the vulnerabilities in the protocols, operating systems, applications and actual physical security of the target system. Readers can easily find several incident news related to D/DoS attacks on the Internet. These attacks are so common that every day it is possible to see them (e.g., please check the digital attack map of Arbor Networks and Google Ideas [3]). It is not hard to predict that IoT will face with D/DoS attacks, either as a target or source of the attacks. In fact, quite recently one of the major Domain Name System (DNS) infrastructure provider of popular web sites and applications was the target of DDoS attacks where a botnet called as *Mirai* compromised thousands of cameras and digital video recorder players [2]. This incident was the first example of IoT being used as an attack source for DDoS. It clearly showed that, protection of IoT networks from attacks is not sufficient and protection of the Internet from IoT networks is needed as well.

A very interesting report [53] on how security of IoT will be playing an important role in defining the cybersecurity of future was published by UC Berkeley Center for Long-Term Cybersecurity in 2016. A group of people from various disciplines developed five scenarios regarding with what will security be like in the future considering various dimensions including people, governments, organizations, companies, society, culture, technological improvements and of course attackers. Although all of the scenarios are related to the security of IoT, the last two scenarios have direct relations. The fourth scenario puts the emphasis on the ubiquity of IoT in a way that IoT will be everywhere and will be playing a vital role on the management of several applications and systems. This will give attackers more chance to target. In such a world, attackers will be able to affect organizations, governments and the daily life of people easier than now. Thus, cybersecurity term will be transformed to just *security* since it will be able to affect everything. The last scenario considers the wearable devices and their novel purpose of use. According to the hypothesis, the wearables of future will not only perform basic measurement tasks, but will be used to track emotional states of humans. Advancements in the technologies will allow such a change.

Emotional, mental and physical state information which is very important for individuals will be the target of attackers and will be used as a weapon against them. Of course in such a scenario, it will be very crucial for people to manage their emotional, mental and physical state and this will affect the society in various ways which we can not imagine. This report clearly shows that if we fail to secure the IoT networks, then the ubiquity and proliferation of IoT will not transform the future to smarter but will cause catastrophic effects on human life, environment, culture and society.

Securing IoT networks is not an easy problem since we have to think of device, network and application characteristics, affordable cryptography-based solutions, physical security of the network and devices, compromise scenarios, intrusion detection systems. Designers and administrators will face many trade-offs, where security will be on one side and cost, network lifetime, Quality-of-Service (QoS), reliability and many more will be on the other side. When we are considering all of these dimensions, we should not avoid the user side. We have to bear in mind that users may not be security-aware. We also have to pay attention to propose user-friendly solutions which consider the usability and the user experience. If our solutions in the services that we provide are not satisfactory, then our efforts will be in vain, making the attackers' job easier.

The goal of this study is to present researchers a comprehensive investigation of IoT security for reliable future IoS. In order to be comprehensive, we analyzed the majority of the digital libraries (i.e., IEEE, ACM, Web of Science, Springerlink, Google Scholar) for quality conference, journal and magazine proposals. Studies published between 2008 and 2017 were included in this work where seventeen studies were analyzed to examine the D/DoS attacks for IoT networks and twenty-six studies were evaluated which either analyze the effects of the attacks, or propose a mitigation or a detection system against such attacks.

The remaining sections of this work are organized as follows: In Sect. 2, we briefly explain the related works. Section 3 explores the characteristics of IoT environments with devices, networks and applications. Section 4 considers Internet of Things security extensively. In Sect. 5, we examine D/DoS attacks for IoT. Section 6 consists of studies which analyze the effects of the D/DoS attacks for IoT networks. In Sect. 7, we examine the mitigation systems against D/DoS attacks, as well as security solutions for specific protocols. Section 8 is on the intrusion detection systems proposed for IoT, where we analyze several proposals from various points of views. In Sect. 9, we discuss the open problems and issues in IoT security and aim to provide new research directions. Finally Sect. 10 concludes this study.

2 Related Works

The Internet of Things is one of the most active topic of research nowadays. There are several surveys which address the security of IoT, attacks, countermeasures and Intrusion Detection Systems for IoT.

Zarpelao et al. [60] proposed a taxonomy of IDSEs based on the placement approaches, detection methods and validation strategies. In their work, the authors

point out that IoT has unique characteristics, which will bring unique threats and novel requirements for IDSes. According to their findings, IDSes proposed for IoT need to address more attacks, more communication technologies and more protocols. They also indicated that IDS traffic should be managed securely and IDS designs should pay attention to the privacy of the host.

Adat et al. [5] proposed a literature review on the security of IoT where history of IoT security, taxonomy of security challenges and requirements, cryptography-based defense mechanisms and IDSes were evaluated. The authors suggested readers to research lightweight authentication schemes, to target 6LoWPAN and RPL security and to consider the resource limitations of IoT devices.

Samaila et al. [46] proposed an extensive analysis of security challenges of IoT. In this study the authors considered several issues including implementation of security in IoT, resource limitations, heterogeneity of IoT environments, applications and devices, security awareness of the users and maintenance of security after deployment.

Yang et al. [58] studied security and privacy issues in IoT. Their work considered the limitations of IoT environments which affect the security and privacy. They provided a classification of the attacks based on the layers of an IoT architecture and analyzed the cryptography-based security solutions for IoT networks in depth.

In this study, we aimed to provide a comprehensive view on security of IoT for reliable IoS. Although there are some topics of interest and points of view in common with the previous reviews, our work tries to depict a more complete picture of security of IoT.

3 Internet of Things

Internet of Things can be defined in several ways from various angles and there is no standard definition for it. However, from the engineering point of view, IoT is a network of any *things*, each supplied with a computing system (i.e., CPU, memory, power source and a communication interface like radio or Ethernet), each is uniquely identifiable and addressable and connected to the Internet. In this section, we will firstly propose a generic architecture for IoT which we think will be helpful to understand IoT environments better. After that, we will summarize the standardized protocol stack [37] we focus on in this study.

3.1 Internet of Things Architecture

We believe that, exploring the architectural components is a very useful way to see the complete picture and understand IoT environments better. In Fig. 1, we outline a generic IoT architecture which is based on the general architectures previously proposed in [24, 57, 59]. The only difference of our architecture from the reference works is that we separated the IoT Access Network Layer from the IoT-Internet Connection Layer, whereas the reference studies combine them into a single layer called either as *Network Layer* or *Transport Layer*.

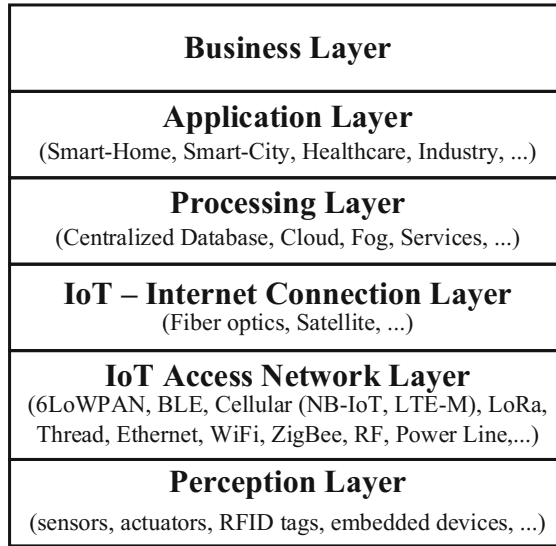


Fig. 1. Generic architecture of IoT

In the generic architecture, the lowest layer is the Perception Layer. It consists of sensors, actuators, RFID tags and any other embedded devices. Most of these devices are expected to be small form-factor devices with constrained resources (i.e., power source, processing, storage and communication interface). The majority of IoT devices will use battery as the power source. However, based on the application environment, mains-powered devices or energy-harvesting elements may exist as well. Since power will be a scarce resource, power consumption of the nodes (i.e., devices in the network) has to be minimized. In addition to various techniques to reduce the power consumption, IoT devices use low-power radios to keep the energy footprint as small as possible and lengthen the network lifetime. Typically low-end microcontrollers with RAM and ROM in the order of KBs constitute the big portion of nodes accommodated in IoT networks. In addition to the resource characteristics, mobility of the devices is important as well. Devices in the Perception Layer can be either static or mobile, but the percentage of mobile devices will be smaller than the static ones.

The IoT Access Network Layer is the second layer in our architecture, in which the nodes in the Perception Layer form a network. In this layer, there are several communication technologies (i.e., 6LoWPAN, Bluetooth Low Energy (BLE), LoRa and LoRaWAN, WiFi, Ethernet, Cellular, ZigBee, RF and Thread) which are candidates for the in-network communication. Most of them are open technologies, whereas some of them are (e.g., ZigBee, LoRa, Cellular) proprietary. These communication technologies provide varying data rates and transmission ranges in return of different power consumptions and costs. Hence, depending on the several design constraints, the nodes in the Perception Layer

can form IoT networks with different characteristics. Among these technologies, BLE, WiFi, LoRa and Cellular offer star-based topologies. However, 6LoWPAN, ZigBee and Thread support mesh topologies, where elements of the network can forward others' packets. Some of them are proposed for specific application areas (i.e., Thread was proposed for smart-home environments). Most of these technologies require a gateway or border router which is used to connect the nodes in IoT network to the Internet.

The third layer in our generic architecture is the IoT - Internet Connection Layer, where a border router or gateway connects the inner IoT network to the Internet via communication technologies, such as fiber optics or satellite communication.

Processing, analysis and storage of the collected data are performed at the Processing Layer. Designers can choose centralized storage and processing systems, or distributed storage and processing systems (e.g., cloud or fog computing environments). Middleware services are provided in this layer based on the processed and analyzed data. This is one of the most important layer in the architecture of IoT, since valuable information is extracted here from the collected data which can be in big volumes, variety and veracity.

The Application Layer is the fifth layer within the generic IoT architecture. In this layer, we see applications in various deployment areas, which make use of the meaningful information obtained from Processing Layer. Applications of IoT can be in home, building, industry, urban or rural environments. Applications of home environments can be health-reporting and monitoring, alarm systems, lighting applications, energy conservation, remote video surveillance [13]. Building environments IoT applications can be Heating Ventilation and Air Conditioning (HVAC) applications, lighting, security and alarm systems, smoke and fire monitoring and elevator applications [31]. Industrial IoT applications can be safety, control and monitoring applications with different emergency classes [38]. In urban environments, there may be broad range of applications. Lighting applications, waste monitoring, intelligent transportation system applications, monitoring and alert reporting are only a few of them. Rural environments may include monitoring applications (e.g., bridges, forests, agriculture, etc.).

The Business Layer is the last layer in the generic architecture, which includes organization and management of IoT networks. Business and profit models are constructed here in addition to charging and management operations [57].

3.2 Standardized Protocol Stack for Low Power and Lossy Networks

Multiple communication technologies exist for the IoT-Access Network Layer as we mentioned in Sect. 3.1. Since Thread, NB-IoT, LTE-M, LoRa/LoRaWAN are very new communication technologies, there were not any studies which focus on the D/DoS attacks that may target such networks during the time we were working on this proposal. ZigBee is a proprietary technology and it also uses the same physical and MAC layers as 6LoWPAN-based networks. Thus PHY and MAC layer attacks for 6-LoWPAN-based networks covers the PHY and MAC

layer attacks for ZigBee-based networks too. WiFi targets more resource-rich devices than 6LoWPAN. Therefore, it may not be a good candidate for low power and lossy networks-based IoT applications where majority of the devices will be battery-powered devices with small form factors and reasonable costs. Bluetooth Low Energy technology might be a good option for the IoT-Access Network Layer with very low power consumption, increased data rate and range. However, it suffers from the scalability problem where Bluetooth-based networks face with issues when the number of slaves exceeds seven [23,25]. Up until Bluetooth 5, park state was supported by Bluetooth which was allowing more than seven slaves to be part of the Bluetooth network in turns. But Bluetooth 5 does not support it any more and instead it brought scatternets, which aims to create multi-hop Bluetooth networks with specific nodes acting as routers between piconets. However, currently no commercial Bluetooth radio supports it and synchronization and routing operations will make the scatternet operation in Bluetooth networks a complex issue to deal with. Hence, considering the aforementioned reasons, we focus on the 6LoWPAN-based IoT networks in this study.

IEEE and IETF proposed several standards and protocols in order to connect resource-constrained nodes to the Internet within the concept of IoT. Palattella et al. [37] proposed a protocol stack for low power and lossy IoT networks which makes use of the protocols/standards proposed by IEEE and IETF. The standardized protocol stack is shown in Fig. 2.

The standardized protocol stack includes IEEE 802.15.4 [1] for physical layer and MAC layers. This standard promises energy-efficient PHY and MAC operations for low power and lossy networks and is also used by Thread and ZigBee technologies.

The expected cardinality of the IoT networks (e.g., of the order of billions) and already exhausted IPv4 address space force IoT to use IPv6 addresses. However, when IPv6 was proposed, low power and lossy networks were not considered, which resulted in the incompatible packet size issue. The maximum transmission unit of IEEE 802.15.4-based networks is far too small compared to IPv6 packet sizes. In order to solve this problem, IETF proposed an adaptation layer,

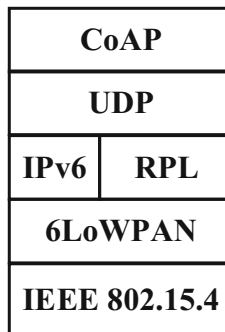


Fig. 2. Standardized protocol stack

IPv6 over Low Power Wireless Personal Area Networks (6LoWPAN) [18]. 6LoWPAN makes use of header compressions to permit transmission of IEEE 802.15.4 fragments carrying IPv6 packets.

RPL [56] was proposed by the IETF as IPv6 routing protocol for low power and lossy networks. Formation of IEEE 802.15.4-based mesh networks was made possible by the RPL routing protocol, which constructs Destination Oriented Directed Acyclic Graphs (DODAG). A DODAG root creates a new RPL instance and lets other nodes to join the network by means of control messages. There are four types of control messages, which are DODAG Information Solicitation (DIS), DODAG Information Object (DIO), Destination Advertisement Object (DAO) and DAO-Acknowledgment (DAO-ACK) messages. DIS messages are broadcasted by new nodes to obtain the information about the RPL instance in order to join the network. Neighbor nodes reply with DIO messages which carry information about the RPL network (i.e., DODAG ID, instance ID, rank, version number, mode of operation, etc.) and their position in the network. The position of a node, which is the relative distance of a node from the DODAG root is named as *rank*. Rank is carried in DIO messages and it is calculated by each node based on the Objective Function (OF) and the rank of neighbor nodes. OF types, include, but are not limited to, hop count, expected transmission count, remaining energy. RPL lets network administrators to select a suitable OF based on the QoS requirements. When a node receives DIO messages from its neighbors, it calculates its rank and informs its neighbors about its rank with a new DIO message. Based on the rank of its neighbors, it selects the one with the lowest rank value as a preferred parent and informs that node with a DAO message. The receiving node replies with a DAO acknowledgment message and thus a parent-child relationship is set up. An example RPL network is shown in Fig. 3.

In RPL upward routes (i.e., the routes towards the DODAG root) are created by means of DIO messages, whereas downward routes are created by DAO messages. In order to minimize the overhead of control messages, RPL uses Trickle

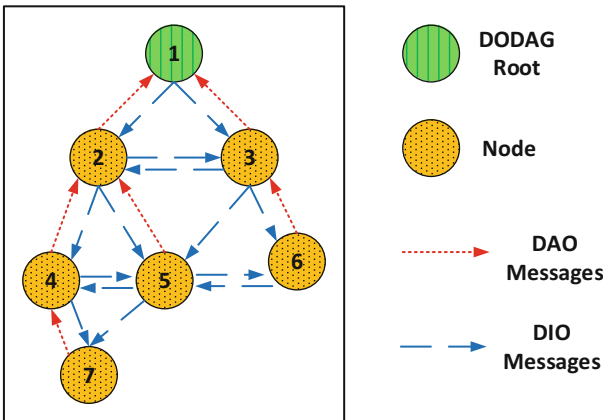


Fig. 3. An example RPL network

Timer [30] to reduce the number of control messages created as network gets more stable. Nodes are expected to follow the rules of the RPL specification in order to create loop-free and efficient RPL DODAGs. In low power and lossy networks, faults and problems tend to occur. To recover from such issues, RPL accommodates repair mechanisms (i.e., global repair and local repair).

The standardized protocol stack for low power and lossy networks employs the Constrained Application Protocol (CoAP) [50] for the application layer. CoAP is built on top of UDP and supports Representational State Transfer (REST) architecture. By means of CoAP, even resource-constrained nodes can be part of the World Wide Web (WWW). In order to optimize the data carried by CoAP messages, the IETF proposed another standard for the binary representation of the structured data called Concise Binary Object Representation (CBOR) [12] on top of CoAP.

4 Internet of Things Security

In Sect. 3.2, we briefly summarized the standardized protocol stack which consists of standards and protocols proposed by IEEE and IETF for low power and lossy networks. In this section, we focus on the security of IoT networks which accommodate the standardized protocol stack.

Securing a communication network is not an easy task and requires a comprehensive approach. In such a study, we have to determine assets, think of threats and consider compromise scenarios and possible vulnerabilities. Following these, we have to find the suitable solutions which will help us to ensure a *secure* system. When we think of the solutions, the first thing that probably comes into our minds is the cryptography. Cryptography promises to provide *confidentiality* and *integrity* of the messages, *authentication* of the users and systems and *non-repudiation* of the transactions. Confidentiality means that the content of the message is kept secret from eavesdroppers. Integrity ensures that the content of the message is not changed and is still the same as the first time it was produced. Authentication allows the end points of the communicating parties to identify each other and determine the correct target of the communication. Non-repudiation prevents one end of the communication to deny its actions that it performs and protects the other end.

In this section, we firstly outline the cryptography-based security solutions for the low power and lossy networks which employ the standardized protocol stack. After that, we analyze the protocols and point out the advantages and disadvantages. Then we will inquire whether cryptography is enough for us or not.

4.1 Cryptography-Based Security Solutions for Low Power and Lossy Networks

A number of cryptography-based solutions exist so as to secure the low power and lossy networks that employ the standardized protocol stack. These solutions are shown in Fig. 4.

CoAP → CoAPs
UDP → DTLS
RPL & IPv6 → IPSec, Secure RPL Control Messages
6LoWPAN →
IEEE 802.15.4 → IEEE 802.15.4 PHY & Link Layer Security

Fig. 4. Cryptography-based security solutions for low power and lossy networks

IEEE 802.15.4 PHY and Link Layer Security [1] provides security for the communication between two neighbors in IEEE 802.15.4-based networks. This hop-by-hop security solution promises confidentiality, authenticity and integrity against insider attackers.

The Internet Protocol Security (IPSec) [22] aims to provide end-to-end security. It consists of a set of protocols, which are Authentication Headers (AH), Encapsulating Security Payloads (ESP) and Security Associations (SA). AH provides authentication and integrity, whereas ESP promises confidentiality in addition to authentication and integrity. Designers can select either of them but regardless of the selection, SA has to run initially to setup the security parameters. IPSec provides security for IP-based protocols and it is independent from the protocols above the network layer.

In addition to IPSec, RPL provides secure versions of the control messages. Although it is optional, confidentiality, integrity and authentication of the control messages are assured.

Datagram Transport Layer Security (DTLS) [44] aims to secure UDP-based applications. Similar to the other solutions, it ensures the confidentiality, integrity and authenticity of datagrams.

CoAP provides security bindings for DTLS in CoAPs scheme. It lets designers to choose to run DTLS with preshared keys, public keys and/or certificates in order to secure CoAP traffic. Although Fig. 4 does not show any other security mechanisms working at the application layer and above, the IETF has draft documents (i.e., Object Security of CoAP (OSCoAP), CBOR Object Signing and Encryption (COSE) and Ephemeral Diffie-Hellman over COSE (EDHOC)) which aim to provide security at the application layer and above.

4.2 Which Security Solution to Use?

As we can see, there are a number of security solutions to protect low power and lossy networks and it is hard to determine which solution to use.

IEEE 802.15.4 PHY and Link Layer Security is independent from the network layer protocols and most of the radios support it. Independence from the

upper layer protocols means that we do not have to change anything with them. However, since IEEE 802.15.4 PHY and Link Layer Security provides the security between two neighbors, trustworthiness of every node on the routing path becomes a very crucial issue. If the routing path has a malicious node, then the security of the messages routed through this path cannot be guaranteed. In addition to this, IEEE 802.15.4 PHY and Link Layer Security works only in the IoT - Access Network Layer in our generic architecture and when messages leave this layer and enter the Internet, they are no more protected [40].

IPSec provides end-to-end security and is independent from the upper layer protocols. End-to-end security guarantees security between two hosts which can be in different networks. Designers do not have to worry about the trustworthiness of the other nodes, devices or networks on the path. However, it brings burden to 6LoWPAN layer, where packets with IPSec require header compression [40]. In addition to this, Security Associations is connection-oriented and simplex, which means if two hosts want to send packets secured with IPSec to each other, then each of them individually need to establish SAs [26]. Furthermore, firewalls may limit the packets with IPSec and Internet Service Providers (ISPs) tend to welcome packets with IPSec as business-class packets and prefer to charge them more. So, if IoT data will be secured with IPSec, there are a number of issues we have to consider before using it.

DTLS serves as the security solution between two UDP-based applications running on different end-points. Although it aims to protect the application layer data, it does not promise security for anything else. This means, if we employ DTLS as the only security solution, then we cannot protect IP headers when packets are passing through the IoT - Access Network Layer and through the Internet. So, security of the routing becomes susceptible to the attacks, such as DoS and DDoS attacks. This is why the primary security concern of DTLS is on D/DoS attacks.

4.3 We Have Cryptography-Based Solutions, Are We All Set?

In Sect. 4.1 we outlined the cryptography-based security solutions very briefly. As we explained in Sect. 4.2, each solution comes with its advantages and disadvantages. It is not an easy task to select the appropriate solution. However, there are a number of other issues which we have to consider when protecting IoT networks.

First of all, cryptography is generally thought to be heavy weight, and is full of resource consuming operations implemented in software and/or hardware. When we consider the resource limitations of the devices in low power and lossy networks, affordability of such solutions becomes questionable. Designers have to face the trade off between security and very crucial parameters such as cost, network life time and performance.

Secondly, although cryptography-based solutions are proved to be secure, proper implementation of the protocols and algorithms is extremely important. However, most of the implementations of these solutions have vulnerabilities as

reported by researchers [8]. In addition to this, in order to shorten the development time, engineers tend to use the code examples shared on forums. These code examples working properly does not mean that they are vulnerability-free [4].

Physical security of the networks and devices are as important as our other concerns. It is directly related to the applicable type of attacks. If the physical security of the deployment area is weak, which is the case for most of the deployments, and if devices do not have protection mechanisms against tampers which is due to reduce the cost, then it is possible for attackers to insert a malicious device or grab a device and extract the security parameters and leave a malicious device back.

In addition to the cost of cryptography, issues with correct implementation and physical security, we have to consider users as well. We know that most of the people are not security-aware and usability of security mechanisms have problems [47]. Therefore, compromise scenarios have to think of users and external people involving with the IoT network, applications and deployment areas.

Although we have cryptography, our networks and systems are still susceptible to some type of attacks, called Denial of Service attacks [54]. In the next section, we will examine the DoS and DDoS attacks which may target low power and lossy networks employing the standardized protocol stack.

5 Denial of Service Attacks Targeting Internet of Things Networks

Denial of Service attacks aim to misuse the available resources in a communication network and degrade or stop the services offered to ordinary users. Since

Table 1. D/DoS attacks which may target IoT networks

Physical layer	MAC layer	6LoWPAN layer	Network layer	Transport and application layer
Node Capture	Jamming	Fragment Dupl.	Rank	Flooding
Jamming	GTS	Buffer Reserv.	Version Number	Desynchronization
Spamming	Backoff Manip.		Local Repair	SYN Flood
	CCA Manip.		DODAG Inconsist.	Protocol Parsing
	Same Nonce		DIS	Processing URI
	Node Spec. Flooding		Neighbor	Proxying and Caching
	Replay Protection		Sybil	Risk of Amplification
	ACK Attack		Sinkhole	Cross-Protocol
	Man-in-the-Middle		Selective Forw.	IP Address Spoofing
	Ping-Pong Effect		Wormhole	
	Boostrapping		CloneID	
	Stenography			
	PANID Conflict			

IoT will be one of the main building block of Internet of Services, detection, mitigation and prevention of such attacks are very crucial.

In this section, we present and explain the D/DoS attacks which may target IoT networks. Table 1 categorizes such attacks with respect to the layers of the standardized protocol stack. This categorization is an extended version of our previous study [10].

5.1 D/DoS Attacks to the Physical Layer

Physical Layer D/DoS attacks are *node capture*, *jamming* and *spamming*.

As its name implies, in *node capture* attacks, attackers capture the physical nodes within the network. The aim of the attackers may be creating routing holes or tampering the device and extracting security parameters. After that, they may place the node back with the compromised software or place the node with a replica of it. By this way, they can apply various attacks (e.g., other attacks categorized as higher layer attacks).

Physical Layer *jamming* attacks comprise of malicious devices creating interference to the signals transmitted in the physical layer [7]. Attackers can constantly, randomly or selectively (i.e., jamming signals carrying specific packets, such as routing or data packets) apply jamming.

In *spamming* attack, attackers place malicious QR codes to the deployment areas which cause users to be forwarded to malicious targets on the Internet [42].

5.2 D/DoS Attacks to the MAC Layer

MAC Layer D/DoS attacks are *link layer jamming*, *GTS*, *backoff manipulation*, *CCA manipulation*, *same nonce attack*, *node specific flooding*, *replay protection attack*, *ACK attack*, *man-in-the-middle*, *ping-pong effect*, *bootstrapping attack*, *PANID conflict* and *steganography*.

Link layer jamming is a type of jamming where frames are jammed instead of signals as in the physical layer [7].

IEEE 802.15.4 standard has an optional feature called as Guaranteed Time Slot (GTS) which works in beacon-enabled operational mode. GTS is intended for timely critical applications that require strict timing with channel access and transmissions. Nodes have to request and allocate time slots in order to use this feature. However, if attackers cause interference during this process (e.g., by jamming), then ordinary nodes cannot register themselves for the guaranteed time slots and thus QoS of the application gets affected. This attack is called *GTS* attack [51].

ACK attack consists of attackers creating interference to Acknowledgment (ACK) frames and thus causing a node to believe that its fragment was not successfully received by the receiving node [7]. By this way targeted nodes are forced to retransmit the same fragment and consume more power. QoS of the running application would be affected by it too. Moreover, it may cause the sender node believe that its next hop neighbor is filtering the messages.

Clear Channel Assessment (CCA) mechanism is used by nodes to sense the channel and find out if any other node is currently using the channel or not. This approach is commonly used to prevent collisions. However, attackers can skip CCA and access the channel, which causes collisions. By this way, delays, retransmissions and unnecessary energy usage occurs. This attack is called *CCA manipulation* [7].

Backoff manipulation attack compromises the backoff periods of Carrier Sense Multiple Access (CSMA)-based medium access with attackers choosing shorter backoff times instead of longer [7]. By this way, they get the chance to use the channel as much as possible and limit the other users' channel accesses.

Sequence numbers are used in the IEEE 802.15.4 standard in order to prevent malicious devices sending the previously sent fragments over and over. However, in *replay protection attacks* [7], attackers can still misuse it by sending frames with bigger sequence number than the targeted ordinary node. This would cause the receiving node drop the fragments coming from the ordinary node since it now looks like it is sending old fragments.

As we mentioned in Sect. 4, IEEE 802.15.4 PHY and Link Layer Security is a candidate security mechanism for IoT security, which promises to protect the communication between two neighbor nodes. If nodes share the same key and nonce values in the implementation of IEEE 802.15.4 PHY and Link Layer Security, then attackers may extract the keys by eavesdropping the messages which happens in the *same nonce* attack [7].

The *PANID Conflict* attack misuses the conflict resolution procedure of IEEE 802.15.4 which functions when two coordinators are placed close to each other in a deployment area and holding the same Personal Area Network ID (PANID). Malicious nodes may transmit a conflict notification message when there is actually no conflict to force the coordinator to initiate the conflict resolution process [7].

Another MAC Layer D/DoS attack is the *ping-pong effect*, where malicious nodes intentionally switch between different PANs [7]. If attackers choose to do it frequently, then they may cause packet losses, delays and extra overhead to the already limited resources.

In the *bootstrapping attack*, attackers aim to obtain useful information about a new node joining the network. In order to do so, firstly a targeted node is forced to leave the network by the attackers. Then when it tries to join the network again, attackers obtain the bootstrapping information which they may use to associate a malicious node to the network [7].

Node specific flooding attacks are a type of flooding attack which is applied at the MAC layer [7]. In this attack, malicious nodes send unnecessary fragments to the target node which aims to consume its resources and thus is no longer able to serve for its ordinary purpose.

The *Stenography* attack abuses the unused fields in the frame format of IEEE 802.15.4. Unused bits can be used by the attackers to carry hidden information [7].

5.3 D/DoS Attacks to the 6LoWPAN Layer

Hummen et al. [19] proposed two attacks, namely *fragment duplication* and *buffer reservation*, which may target the 6LoWPAN Adaptation Layer.

In *fragment duplication* attack, attackers duplicate a single fragment of a packet and thus force the receiving node to drop the fragments of the corresponding packet. In this attack, attackers abuse the approach of 6LoWPAN standard which deals with the duplicate fragments. The standard advises to drop the fragments of a packet in case of duplicates so as to get rid of the overhead of dealing with duplicates and save resources. However, malicious nodes can turn this naive mechanism into a DoS attack very easily.

In *Buffer reservation* attacks, attackers reserve the buffer space of the targeted node with incomplete packets and keep it occupied as long as possible. Since resources are limited, nodes cannot afford to spare extra buffer space for the incomplete packets of other nodes. Thus, during the time the attacker holds the buffer space, ordinary nodes' fragments cannot be accepted. Readers should note that, this behavior of 6LoWPAN is possible when 6LoWPAN is configured to forward the fragments according to the route-over approach, where all fragments of a packet are reassembled by the receiving node before being forwarded.

5.4 D/DoS Attacks to the Network Layer

D/DoS attacks which may target the IoT Network Layer can be divided into two categories: RPL-specific and non-RPL-specific attacks. RPL-specific attacks are *rank*, *version number*, *local repair*, *DODAG inconsistency* and *DIS* attacks. Non-RPL-specific attacks are the ones which are already known from the wireless sensor networks, and other communication networks research. Although they look old-fashioned, they are still applicable in RPL-based networks. Non-RPL-specific attacks are *sybil*, *sinkhole*, *selective forwarding*, *wormhole*, *cloneID* and *neighbor* attacks.

RPL-Specific Attacks. D/DoS attacks which may target RPL networks abuse the vulnerabilities of the RPL protocol design. RPL, designed by the IETF for the routing of IPv6 packets on low power and lossy networks, has vulnerabilities with the control plane security and attackers can easily misuse it. In order to secure RPL networks, the IETF advises to use cryptography-based security solutions, secure control messages and some attack-specific countermeasures (e.g., using location information, multi-path routing) [52]. However, as explained in Sect. 4.3, there are several issues to consider with security and it is highly probable that RPL-based networks will be susceptible to D/DoS attacks.

Rank is a very crucial parameter of the RPL protocol which represents a node's position within the DODAG. This position is a relative distance of a node from the DODAG root. The distance is determined with respect to the Objective Function and can be based on the hop count, link quality, remaining power etc. Rank is used to create an efficient DODAG according to the application needs and to set up the child-parent relationship. For optimized and loop-free

DODAGs, nodes have to follow the rules. However malicious nodes may use rank in various ways to apply D/DoS attacks. In [27] and [28], an attacker node selects the neighbor with worst rank as a preferred parent instead of choosing the one with the best rank. Thus an inefficient DODAG is created which causes delays and an increased number of control messages. In [29], the attacker intentionally skips applying the rank check which breaks the rank rule constructing the loop-free parent-child relationship.

RPL has two repair mechanisms in order to keep the DODAG healthy. One of them is the global repair operation where the entire DODAG is re-created. According to the RPL specification, only the DODAG root can initiate the global repair mechanism by incrementing the *Version Number* parameter. Every DODAG has a corresponding version number that is carried in DIO messages. When the root increments the version number, nodes in the RPL network find out the global repair operation by checking the version number in the incoming DIO messages. They exchange control messages and setup the new DODAG. However, there is no mechanism in RPL which guarantees that only the DODAG root can change the version number field. Malicious nodes can change the version number and force the entire network to set up the DODAG from scratch [11, 35]. This attack is called *Version Number* attack and it affects the network with unnecessary control messages, delays, packet losses and reduced network lifetime.

Similar to the global repair, the local repair mechanism of RPL can be the target of a D/DoS attack called *local repair* [27, 29]. Local repair is an alternative repair solution of RPL which aims to solve the local inconsistencies and issues and cost less than the global repair mechanism since it involves a smaller portion of the network. If nodes find out inconsistencies (e.g., loops, packets with wrong direction indicators), then they start the local repair mechanism which consists of exchanging control messages and re-creating the parent-child relationships and getting appropriate ranks again. However malicious nodes can start local repair when there is no need so as to misuse the resources. This type of attack is called *local repair* attack.

RPL has a data path validation mechanism, in which headers of the IPv6 data packets carry RPL flags that indicate the direction of the packet and possible inconsistencies with the rank of the previous sender/forwarder. When a node receives a packet with those flags indicating an inconsistency, it drops the packet and starts the local repair mechanism. In *DODAG inconsistency* attacks [49], attackers can set the corresponding flags of a data packet before they forward it and force the receiver node to drop the packet and start local repair.

The last D/DoS attack specific to RPL is the *DIS* attack. DIS messages are used in RPL when a new node wants to join the network and therefore asking for information about the RPL network. Attackers can send unnecessary DIS messages in *DIS* attacks [27], which causes the neighboring nodes to reset their DIO timers and send DIO messages frequently. Thus, the attacker forces nodes to generate redundant control messages and consume more power.

Attacks not Specific to RPL. Attacks which are not specific but still applicable to RPL are *neighbor*, *sybil*, *sinkhole*, *selective forwarding*, *wormhole* and *cloneID*.

A malicious node can apply the *neighbor* attack by retransmitting the routing control messages it hears [27]. This behavior causes neighbor nodes to think that the source of the control message is close to them and take actions accordingly. Actions could be sending control messages back, trying to select it as a preferred parent, etc. If the attacker uses a high power radio, then it may affect a large portion of the network by this way.

In *sybil* attacks, a malicious node seems to act as multiple nodes, introducing itself with multiple logical identities [54]. If there is a voting mechanism running in the IoT network (e.g., voting based security mechanisms, cluster head selection), attackers can apply sybil to change the results and thus take control of the complete network or a portion of the network.

The *CloneID* attack is similar to the sybil attack but works in a different dimension. The attacker in this case places the clones of a malicious node or normal node to the multiple positions at the network [41, 54]. This attack has similar aims as sybil and it may also be called *node replication* attack.

Sinkhole attacks are another type of attacks where malicious nodes advertise good routing parameters to show themselves as candidate parents. In RPL, attackers can advertise good ranks, which causes the neighbor nodes to select it as the preferred parent [41, 54, 55]. When a malicious node is selected as the preferred parent by neighbor nodes, then it can apply other attacks, such as selective forwarding.

In *selective forwarding* attacks, a malicious node inspects the incoming packets, drops the ones it is interested in and forwards the rest [41, 54]. For example, it may forward only the routing messages, whereas it may drop the data packets. Or, malicious node may filter specific packets sourced from or destined to specific addresses.

The last attack we explore in this category is the *wormhole* attack [39, 54]. In wormhole attacks, at least a couple of malicious nodes create a hidden communication channel by means of multiple radios and transfer the overheard messages transmitted at one end point to another. This may work bidirectional as well. By this way, two sets of nodes around each attacker believe that they are in the communication range of each other, which causes several issues.

5.5 D/DoS Attacks to the Transport and Application Layer

D/DoS attacks which may target Transport and Application Layers are *flooding*, *desynchronization*, *SYN flood*, *protocol parsing*, *processing URI*, *proxying and caching*, *risk of amplification*, *cross-protocol* and *IP address spoofing* attacks [21, 50]. The majority of the attacks mentioned here were not studied in the literature and the IETF considers them as possible threats for CoAP.

6 Studies that Analyze D/DoS Attacks for Internet of Things

The previous section was about the possible D/DoS attacks which can target the IoT networks. Starting from this section, we will analyze the studies for the aforementioned attacks. In this section, we will explore the works which investigate the effects of the attacks.

Sokullu et al. [51] proposed GTS attacks to IEEE 802.15.4 in 2008. In their work, they also analyzed the effects of the attack in the bandwidth utilization of Contention Free Period (CFP). They considered single and multiple attackers where attackers can either attack randomly or intelligently. They found out significant decrease in the bandwidth utilization of CFP periods due to GTS attacks.

Le et al. analyzed the rank attack in [28] in RPL networks in 2013. In this work, they applied the rank attacks with different cases where the attacker constantly applies the attack or switches between legitimate and malicious behaviors frequently. Analysis with respect to combinations of attacking cases show that if the rank attack is applied in a dense part of the network, then its effect is more detrimental. They also realized that, the number of affected nodes, number of generated DIO messages, average end-to-end delay and delivery ratio can be the indicator of such attacks.

Mayzaud et al. studied RPL version number attacks in [35] in 2014. Their investigation with a single attacker in a grid topology at varying positions showed that the location of the attacker is correlated to the effects of the attack. If the attacker is located far from the DODAG root within the grid, then its effect is larger than when attacker is closer to the root.

The Version number attack is analyzed by another work [11] proposed by Aris et al. in 2016. In this study, the authors considered a factory environment consisting of varying topologies (i.e., grid and random) with different node mobilities (e.g., static and mobile nodes). A probabilistic attacker model is incorporated here. Based on the simulations, in addition to the location-effect correlation found in Mayzaud's work [35], the authors found out that the mobile attackers'

Table 2. Categorization of the studies that analyze the D/DoS attacks for IoT

Proposal	Target attack	Finding
Sokullu [51]	GTS (MAC Layer)	Significant bandwidth utilization decrease in CFP
Le [28]	Rank (Routing)	Dense networks are more vulnerable
Mayzaud [35]	Version Number (Routing)	<i>Attacking position-effect of the attack correlation</i>
Aris [11]	Version Number (Routing)	Mobile attackers are more detrimental and attack triples the power consumption of the network

effect can be as detrimental as the farthest attacking position in the network. They also showed that, version number attacks can increase the power consumption of the nodes by more than a factor of two.

Table 2 categorizes the studies which analyze the effects of the D/DoS attacks for IoT. When we review the studies in this section, we realize that researchers focused on the IoT-specific attacks rather than the attacks which we are already familiar with from the Wireless Sensor Networks research (i.e., selective forwarding, wormhole, sinkhole, etc.). In addition to this, three of the studies found out correlations with the success of the attack and the attack settings. Such findings can be extremely useful in defending IoT networks against the attackers and designing better detection and mitigation systems which consider these findings. In Table 1, we had provided a categorization of the D/DoS attacks for IoT and it is clear that many attacks have not been implemented and analyzed in a similar manner.

7 Mitigation Systems and Protocol Security Solutions for Internet of Things

Mitigation systems are proposed by researchers in order to minimize the effects of the attacks. Such systems are far from being a complete security solution but still can increase the strength of the system against attackers. In this context, existing protocols are enriched with additional features by the designers which can mitigate the detrimental effects of the D/DoS attacks. On the other hand, protocol security solutions referred here consist of mechanisms which aim to secure a communication protocol or a specific part of it. In this section, readers can find the studies which either propose a security solution or mitigate the effect of the attacks.

Dvir et al. proposed VeRA [16], a security solution for the crucial version number and rank parameters carried in DIO messages in 2011. Their solution makes use of hash chains and message authentication codes in order to securely exchange these RPL parameters in DIO messages.

Weekly et al. [55] evaluated the defense techniques for sinkhole attacks in RPL in 2012. They compared a reduced implementation of VeRA to their novel technique called as Parent Failover. Parent Failover uses Unheard Node Set which includes the IDs of the nodes that the BR did not hear from. Each node blacklists its parent if it sees itself in the list in this technique.

Wallgren et al. [54] proposed implementations of routing attacks (i.e., selective forwarding, sinkhole, hello flood, wormhole, sybil) which are not specific to RPL. They did not analyze the effects of the attacks. However, they made comments on possible mitigation/detection mechanisms against such attacks. Their mitigation ideas include usage of geographical location information, incorporation of cryptography schemes, using multiple routes and/or RPL instances and keeping track of the number of nodes within the network. Although the authors suggest to use such mechanisms against the corresponding attacks, they did not implement the mitigation mechanisms and analyze the performance of them.

Hummen et al. [19] proposed two novel attacks to 6LoWPAN adaptation layer, which are fragment duplication attack and buffer reservation attack. They also proposed two novel mitigation mechanisms against these attacks. For fragment duplication attacks, the authors proposed hash chain structures which create a binding for fragments of a packet to the first fragment of the corresponding packet. In order to mitigate the effects of buffer reservation attacks, they suggested to split the reassembly buffer into fragment-sized slots and let multiple fragments belonging to different packets use it. They merged split buffer approach with a fragment discard mechanism in case of overloaded buffer conditions.

In 2014, Sehgal et al. [49] proposed a mitigation study which targets DODAG inconsistency attacks. According to the authors, RPL uses a threshold to mitigate the effects of such an attack. In RPL, a node receiving a data packet with flags indicating an inconsistency drops the packet and resets its trickle timer. A node can do this until reaching a threshold. After this threshold it does not reset the trickle timer any more. This proposal changes the constant threshold of RPL to an adaptive threshold to mitigate the effects of the attack better.

Another mitigation technique for DODAG inconsistency attacks was proposed by Mayzaud et al. [33] in 2015. It is an improved version of the mitigation technique proposed in Sehgal's work [49]. In the former study, packets with 'R' flags set were counted, whereas in this study, the number of trickle timer resets are counted. Based on this, a node either drops the packets and resets trickle timer, or forwards the packets with modifying the R and O flags to the normal state.

Table 3. Categorization of the mitigation systems and protocol security solutions

Proposal	Target attack	Mitigation/Security mechanism
VeRA [16]	Rank and Version Number (Routing)	Hash chains and Message Authentication Codes
Weekly [55]	Sinkhole (Routing)	Reduced VeRA and Unheard Node Set
Wallgren [54]	Routing attacks not specific to RPL	Geographical Location Info., Cryptography, Multiple Paths and Instances, Cardinality of the Network
Hummen [19]	Fragment Duplication and Buffer Reservation (MAC)	Content Chaining Using Hash Chains, Split Buffer with Fragment Discard
Sehgal [49]	DODAG Inconsistency (Routing)	Adaptive Threshold for Inconsistency Situations
Mayzaud [33]	DODAG Inconsistency (Routing)	Adaptive Threshold for Inconsistency Situations
Ramani [43]	CloneID (Routing), General DoS	Distributed Firewall

In 2016, Ramani proposed a two-way firewall [43] for low power and lossy networks. The two-way firewall analyzes the traffic destined to the 6LoWPAN network and traffic leaving from the network. The proposed firewall was tested against the CloneID and simple DoS attacks. The main module of the firewall works on the BR and becomes active when packets destined to the CoAP and DTLS ports are captured. Packets are parsed into incoming and outgoing packets and their IP addresses and ports are verified. After this check, information related to the packet is saved and checked against the protocol rules. Erroneous packets are dropped here. Also the nodes in the 6LoWPAN network are equipped with mini-firewall modules which inform the main firewall about their behavior.

Table 3 categorizes the Mitigation Systems and Protocol Security Solutions for IoT. Mitigation mechanisms against routing attacks constitute the majority of the studies in this section. Researchers targeted both RPL-specific attacks and other routing attacks which can be applied to RPL as well. Considering the resource limitations in IoT networks, we can see that three proposals use hash functions as lightweight solutions.

8 Intrusion Detection Systems for Internet of Things

In this section, we will survey the literature for intrusion detection systems proposed against D/DoS attacks for IoT networks. This section is organized as follows: Firstly, we will briefly give some background information about Intrusion Detection Systems (IDS). After that, we will analyze the IDSes proposed for IoT.

8.1 Intrusion Detection Systems

Intrusion Detection Systems serve as a strong line of defense for computer networks against the attackers. Without IDS, the puzzle of a *secure* network is incomplete. As explained in Sect. 4.3, despite having cryptography-based solutions, attacks are still possible and IDS comes into the picture here, where it monitors and analyzes the traffic, data, behavior or resources and tries to protect the network from attackers.

Intrusion Detection Systems can be explored from various points of view. Figure 5 shows a 3D Cartesian Plane of IDSes, where axes depict important categories which may be helpful to classify the IDSes. Although not shown, there may be other dimensions in this figure, such as operation frequency and targeted attacks.

Intrusion Detection Systems: Detection Techniques. IDSes can be divided into four classes based on the detection technique. These are *anomaly-based*, *signature-based*, *specification/rule-based* and *hybrid* systems where the former two are the most popular ones.

Anomaly-based systems learn the behavior of the system when there is no attack and create a *profile*. Deviations from the profile show possible anomalies. Anomaly-based IDSes can detect the new attacks since attacks are expected

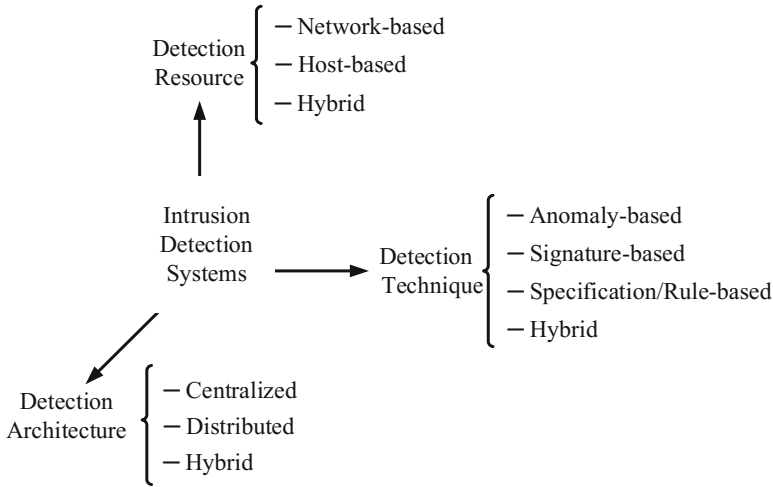


Fig. 5. Intrusion detection systems

to cause deviations from the ordinary behavior. However, they can create false alarms and incorrectly classify legitimate connections as intrusion attempts. In addition to this, anomaly-based techniques are generally believed to be more complex and to use more resources than the other detection techniques.

Signature-based systems aim to detect intrusions by making use of attack signatures/patterns. Typically signatures are stored in a database and IDS tries to match them when analyzing the connections, packets or resources. If the database does not have a signature for an attack, which happens in case of new attacks, such systems cannot detect it. Otherwise they promise high detection rates for the known attacks and they do not suffer from false alarms. If we use signature-based techniques, we have to consider how to deal with new attacks since our IDS will probably skip them. Also we have to think about the storage cost of the signatures.

Specification/rule based systems require specifications of the protocols/systems and create rules based on the specifications. These rules separate legitimate connections from the malicious ones. In such systems creation of the specification and coverage of the created rules are important issues which affect the performance of the IDS.

Hybrid intrusion detection systems consider advantages and disadvantages of the previous three detection techniques and aim to benefit from multiple of them at the same time. Of course such a decision may be costly in terms of the available resources.

Intrusion Detection Systems: Detection Resources. Intrusion Detection Systems can be divided into three categories in terms of the resources they use for detection. These are *network-based*, *host-based* and *hybrid* detection systems.

Network-based IDSes use the incoming and outgoing monitoring traffic to/from the network in addition to the internal traffic to detect the intrusions. Network-based IDSes can have a global view of the network and use it to boost the detection performance. However, such systems lack the information about the individual resource consumptions and logs of the nodes within the network which may be crucial for the detection of specific attacks.

Host-based intrusion detection systems consider the traffic only coming to and leaving from the host. Such systems monitor the resources and logs of the hosts as well which may provide hints about attacks. Since they work locally, they cannot have a global knowledge about the state of other nodes or the network which can be very useful to increase the performance of the IDS.

Hybrid IDSes combine the strengths of network-based and host-based systems that benefit from both network and node resources.

Intrusion Detection Systems: Detection Architecture. Architecture of Intrusion Detection Systems can be *centralized*, *distributed* or *hybrid*.

Centralized IDSes place the intrusion detection to a central location and all of the monitoring information has to be collected here. One of the main reasons to select a central point for intrusion detection can be the available resources. As mentioned previously, anomaly detection techniques can be resource-hungry and it may not be feasible to place them on every node due to resource-constraints. Therefore, a resource-rich node, such as border router, can accommodate the intrusion detection system. However, centralized systems come along with communication overhead since monitoring data has to be carried all the way to the central location. If malicious nodes prevent monitoring data from reaching to the centralized IDS, then they may achieve to mislead the IDS.

In *Distributed* IDSes, intrusion detection runs locally at every node in the network. In order to afford an IDS at every node, designers have to tailor the detection technique or algorithm according to the available resources. This approach clearly does not have any communication overhead, however the IDS has only local information to analyze in order to detect the intrusions.

Hybrid IDSes again harmonize both of the detection architectures and try to benefit from each of them as much as possible. In such systems, IDS is divided into modules and these modules are distributed along the network. Modules at every node can apply intrusion detection to a certain extend, may share less information (in comparison with centralized IDSes) with the centralized module and thus both reduce the communication overhead and enjoy the rich resources of the centralized module.

8.2 Intrusion Detection Systems Proposed for Internet of Things

Cho et al. [15] proposed a botnet detection mechanism for 6LoWPAN-based networks in 2009. They assumed that the nodes in the IoT network use TCP transport layer protocol. Nearly seven years before the Mirai botnet, this study considered how IoT networks can be used as a botnet for DDoS attacks.

The authors thought that, if there exists a malicious node on the forwarding path, then it can forge the packets and direct them to the target victim address based on the command of the bot master. A detection mechanism is placed at the 6LoWPAN gateway node which analyzes the TCP control fields, average packet lengths and number of connections to detect the botnets. The idea is based on hypothesis that the ordinary IoT traffic should be very homogeneous and botnet would cause significant deviations on the traffic.

Le et al. proposed a specification-based IDS [29] for IoT in 2011. It targets rank and local repair attacks. Their work assumes that a monitoring network is set up at the start of the network with minimum number of trustful monitoring nodes which has full coverage of the RPL network and has capability to do additional monitoring jobs. In this context, it has a distributed architecture and it is a network-based IDS. Each Monitoring Node stores the IDs, ranks and preferred parents of neighboring nodes. MNs accommodate a Finite State Machine (FSM) of RPL with normal and anomaly states to detect the attacks. If a MN cannot decide whether a node is an attacker or not, then it can ask the other MNs. This IDS was not implemented and the authors did not specify the format of the communication between the monitoring nodes.

Misra et al. [36] proposed a learning automata based IDS for DDoS attacks in IoT. When there is an attack taking place, packets belonging to the malicious entities need to be sampled and dropped. This study aims to optimize this sampling rate by means of Learning Automata (LA). Firstly DDoS attacks are detected at each IoT node in the network based on the serving capacity thresholds. When the source of the attack is identified, all of the nodes are informed about it. In the next step, each node samples the attack packets and drops them. This is when the LA solution comes to the scene. Sampling rate of the attack packets are optimized by means of the LA.

SVELTE IDS [41] was proposed by Raza et al. in 2013. It targets sinkhole and selective forwarding attacks. It has a hybrid architecture where it places lightweight IDS modules (i.e., 6LoWPAN mapper client and mini firewall module) at the resource-constrained devices and the main IDS (i.e., 6LoWPAN mapper, intrusion detection module and distributed mini firewall) at the resource-rich Border Router (BR). 6LoWPAN Mapper at the BR periodically sends requests to the mapper clients at nodes. Mapper clients reply with their ID, rank, their parent ID, IDs of neighbors and their ranks. Based on the collected information about the RPL DODAG, SVELTE compares ranks to find out inconsistencies. It also compares the elements of the white-list and elements of current RPL DODAG and uses nodes' message transmission times to find out the filtered nodes. The mini firewall module is used to filter outsider attackers. In addition to these, nodes change their parents with respect to packet losses encountered. In terms of the detection resources used, we can classify SVELTE into network-based IDSes. We can also put it into the category of specification/rule-based IDSes.

In the same year with SVELTE, Kasinathan et al. proposed a centralized and network-based IDS [21] and its demo [20] for 6LoWPAN-based IoT networks. The motivation of this study is the drawback of centralized IDSes which

suffer from internal attackers. As mentioned in Sect. 8.1, internal attackers may prevent monitoring data from reaching the centralized IDS. This is due to the fact that monitoring data is sent through the shared wireless medium, which can be interfered by attackers. In this IDS architecture, the authors place monitoring probes to the IoT network which have wired connections to the centralized module. Evaluation of their architecture was done via a very simple scenario where they used an open source signature-based IDS. A monitoring probe sends monitoring data under the UDP flood attack.

Amaral et al. [6] proposed a network-based IDS for IPv6 enabled WSNs. In the proposed scheme, watchdogs which employ network-based IDS are deployed in specific positions within the network. These nodes listen their neighbors and perform monitoring of exchanged packets. IDS modules at each watchdogs use rules to detect the intrusion attempts. These rules are transmitted to watchdogs through a dedicated channel. In order to dynamically configure the watchdogs, the authors used policy programming approach.

In 2015, Pongle et al. proposed an IDS [39] for wormhole attacks. Their IDS has a hybrid architecture similar to SVELTE. Main IDS is located at the BR and lightweight modules are located at the nodes. This study assumes that the nodes are static and the location of each node is known by the BR at the beginning. The main IDS collects neighbor information from nodes and uses it to find out the suspected nodes whose distance is found to be more than the transmission range of a node. The probable attacker is detected by the IDS based on the collected Received Signal Strength Indicator (RSSI) measurements related to the suspected nodes. In terms of the detection resource, we can consider this study as a network-based IDS. And from the detection technique point of view, it can be counted as a specification/rule-based IDS.

Another IDS proposed in 2015 was INTI [14] which targets sinkhole attacks. INTI consists of four modules. The first module is responsible for the cluster formation, which converts the RPL network to a cluster-based network. The second module monitors the routing operations. The third module is the attacker detection module, where reputation and trust parameters are determined by means of Beta distribution. Each node sends its status information to its leader node, which in turn determines the trust and reputation values. Threshold values on these parameters define whether a node is an attacker or not. The fourth module isolates the attacker by broadcasting its information. INTI can be classified as an anomaly-based IDS with distributed IDS architecture. In terms of the detection resources, we can put it into the category of hybrid IDSes.

Sedjelmaci et al. proposed an anomaly-detection technique [48] for low power and lossy networks in 2016. Unlike the other IDSes targeting specific attacks and aiming to detect them, this study focuses on the optimization of running times of detection systems. The motivation of the study is derived from the fact that anomaly-based systems require more resources compared to signature-based systems. If our system can afford to be a hybrid system, having both of the detection systems, then we have to optimize the running time of the anomaly-detection module in order to lengthen the lifetime of the network. The authors

choose game-theory in this study for the optimization of the running time of the anomaly detection system. They claim that, thanks to the game-theory, anomaly detection runs only when a new attack is expected to occur. Anomaly detection runs only during such time intervals and create attack signatures. The signature-based system in turn puts this signature to its database and runs more often than the anomaly-detection system.

Mayzaud et al. proposed a detection system [32] for version number attacks in 2016. Their system uses the monitoring architecture which was proposed in the authors' earlier work [34]. Their monitoring system makes use of the multiple instance support of RPL protocol. It consists of special monitoring nodes with long range communication radios. These nodes are assumed to be covering the whole network and can send the monitoring information to the DODAG root using the second RPL instance that was setup as the monitoring network. In the proposed IDS, monitoring nodes eavesdrop the communication around them and send the addresses of their neighbors and addresses of the nodes who sends DIO messages with incremented version numbers to the root. The root detects the malicious nodes by means of the collected monitoring information. However, the proposed technique suffers from high false positives. This IDS can be counted as a network-based IDS with centralized detection architecture. We can also categorize it as a specification/rule-based IDS.

Another IDS proposed in 2016 was Saeed et al.'s work [45]. This study focuses on the attacks targeting a smart building/home environment where readings of sensors are sent to the server via a base station. In this study, the focus is on the attacks that target the base station. These attacks include software-based attacks and other attacks (i.e., performance degradation attacks, attacks to the integrity of the data). The anomaly-based with a centralized architecture is located at the base station. It consists of two layers. The first layer is responsible for analyzing the behavior of the system and detecting anomalies. It uses Random Neural Networks to create the profile and detect the anomalies. The second part is responsible from the software-based attacks. It comes up with a tagging mechanism to pointer variables. Accesses with the pointer are aimed to be limited with respect to the tag boundaries.

Le et al. proposed a specification-based IDS [27] which is based on their previous work [29]. Their IDS targets rank, sinkhole, local repair, neighbor and DIS attacks. Firstly the proposal obtains an RPL specification via analysis of the trace files of extensive simulations of RPL networks without any attacker. After the analysis of the traces for each node, states, transitions and statistics of each state are obtained. These are merged to obtain a final FSM of RPL which helps them to find out instability states and required statistics. This study organizes the network in a clustered fashion. The IDS is placed at each Cluster Head (CH). CH sends requests to cluster members periodically. Members reply with neighbor lists, rank and parent information. For each member, CH stores RPL related information. CH runs five mechanisms within the concept of IDS. These mechanisms are understanding the illegitimate DIS messages and checks for fake DIO messages, rank inconsistencies and rules, and instability of the network. CH makes use of three thresholds to find out the attackers. These are number

of DIS state and instability state visits, and number of faults. This IDS can be counted as network-based IDS with a distributed architecture.

Aris and Oktug proposed a novel IDS design [9] in 2017 which is an anomaly-based IDS with hybrid architecture. In this study lightweight monitoring modules are placed at each IoT nodes and the main IDS is placed at the BR. Monitoring modules send RPL-related information and resource information of the node. The main IDS module periodically collects the monitoring information and also works as a firewall, where it can analyze the incoming and outgoing traffic from and to the Internet. In this study, each IDS module working on different RPL networks can share suspicious events information with each other. Each IDS works autonomously and detects anomalies using the monitoring information of 6LoWPAN network, firewall information and suspicious events information. When anomalies are detected, nodes within the network are informed via white-lists, whereas other IDSes are informed via the exported suspicious events information. This anomaly-based IDS is a hybrid IDS in terms of architecture and the detection resources used.

Table 4 categorizes the IDSes for IoT. This table shows that majority of the systems are specification/rule-based. It clearly shows that, researchers focused on the protocols (i.e., RPL) rather than a common approach of creating a profile of normal behavior. This observation is also related to signature-based systems being rarely proposed for IoT. The reason may be due to the hardness of creating the signatures for the aforementioned attacks in IoT environments. In terms of the detection architecture, we can see that researchers consider every possible architecture and there is no outperforming option here. When we analyze the detection resources used, most of the studies are network-based IDSes. This shows that, node resources and logs are not yet used frequently by IoT security

Table 4. Categorization of intrusion detection systems for IoT

Proposal	Det. arch	Det. technique	Det. resource
Cho [15]	Centralized	Anomaly-based	Network-based
Le [29]	Distributed	Specification/Rule-based	Network-based
Misra [36]	Distributed	Specification/Rule-based	Network-based
SVELTE [41]	Hybrid	Specification/Rule-based	Network-based
Kasinathan [21]	Centralized	Signature-based	Network-based
Amaral [6]	Distributed	Specification/Rule-based	Network-based
Pongle [39]	Hybrid	Specification/Rule-based	Network-based
INTI [14]	Distributed	Anomaly-based	Hybrid
Sedjelmaci [48]	Distributed	Hybrid	Host-based
Mayzaud [32]	Centralized	Specification/rule-based	Network-based
Saeed [45]	Centralized	Anomaly-based	Network-based
Le [27]	Distributed	Specification/rule-based	Network-based
Aris [9]	Hybrid	Anomaly-based	Hybrid

Table 5. Target attacks & Implementation environments of intrusion detection systems for IoT

Proposal	Target attacks	Implem. env.
Cho [15]	Botnets	Custom Simulation
Le [29]	Rank, Local Repair	Not-implemented
Misra [36]	General DoS	Custom Simulation
SVELTE [41]	Sinkhole, Selective-forwarding	Contiki Cooja
Kasinathan [21]	UDP flooding	Project Testbed
Amaral [6]	General DoS	Project Testbed
Pongle [39]	Wormhole	Contiki Cooja
INTI [14]	Sinkhole	Contiki Cooja
Sedjelmaci [48]	General DoS	TinyOS TOSSIM
Mayzaud [32]	Version Number	Contiki Cooja
Saeed [45]	Software-based attacks, Integrity attacks, Flooding and other	Prototype impl.
Le [27]	Rank, Sinkhole, Local Repair, Neighbor, DIS	Contiki Cooja

researchers. This may be due to the already limited resources of the nodes which may already be used 100% (e.g., RAM) or no space to store logs. But it is interesting to see that no proposal considers to use the deviation of the power consumption as an intrusion attempt.

Table 5 compares the studies in terms of target attacks and implementation environments. The majority of the attacks targeted by IDSes for IoT are routing attacks as shown in the table. A big portion of the studies focus only on a single attack, whereas only a few studies consider multiple attacks. When we consider these attacks, nearly all of them are insider attacks. This means, IoT security researchers in this concept are not thinking of the threats sourced from the Internet yet. In addition to this, only one study targets software-based attacks. However, we know that embedded system developers choose programming in C language, which may open software-based vulnerabilities to the attackers targeting IoT. In terms of the implementation environment, Contiki Cooja is the environment selected by most of the researchers.

9 Discussion and Open Issues

In this study, we provided an extensive overview of Internet of Things security in order to ensure reliable Internet of Services for the future. Of course there may be other studies which were left unmentioned unintentionally. Considering the limitations, attacks, cryptography-based security solutions and studies in the literature, there are still several issues to research in order to reach a secure IoT environment.

One of the major points to consider is the usability and user experience when providing security to IoT environments. We have to consider users and provide user-friendly schemes which will not disturb the satisfaction of users while promising security. This is directly related to the success and acceptance of our solutions. Otherwise, our efforts will be in vain, making the attackers' job easier.

As we have mentioned in Sect. 5.2, some of the MAC layer attacks make use of jamming attack to reach their aim. If we find a solution against jamming attacks, then this may make it easier to mitigate the effects of such attacks.

IDSes proposed for IoT use thresholds to decide whether a node/connection is malicious nor not. However, considering the proposals, thresholds seem to be set intuitively, not based on a scientific technique. This approach clearly limits the applicability and reproducibility of the proposed mechanism. The way we set thresholds may be an important issue to think about when designing IDSes.

Assumptions of the studies are another point to re-consider. Some studies assume that there is a monitoring network covering the whole network with a minimum number of nodes and was setup at the beginning and is ready to use. Such assumptions have to be supported with deployment scenarios, otherwise it may not be realistic to have such assumptions.

Anomaly-based and also specification-based IDSes typically require an attack-free period where the underlying system will be able to understand the normal operating conditions and create a profile accordingly. However, this may not be possible for real-life deployments. In addition to this, if our deployment includes thousands of nodes, then ensuring such a period may not even be feasible.

Most of the studies target only a small number of attacks as mentioned in another study [60]. Researchers have to target a broader range of attacks or propose systems which have the capability to be extended to detect other attacks too.

In terms of the types of attacks, most of the studies focus only on insider attacks, whereas outsider attacks from the Internet have to be researched and analyzed. When we consider Table 1, we can see that attacks above the network layer were not studied extensively. This clearly shows that transport and application layers of IoT may be vulnerable to attacks and IoT will be mentioned a lot within news of DDoS attacks.

Another issue with IoT security research is related to reproducibility and comparability of the studies. When we have a look at the studies, most of the authors keep the source codes of their implementations closed. In addition to this, IoT security research does not have datasets which can be used by the researchers as a common performance evaluation benchmark although testbeds that are publicly available exist. It would enrich the IoT security research if more researchers share their implementations with public and organizations provide datasets which can be used for evaluation purposes.

10 Conclusion

In this study while we aim to provide a comprehensive overview of security of IoT for reliable IoS, we incorporated the points of view that include unique characteristics of IoT environments and how they affect security, architectural components

of IoT and their relation to the standardized protocol stack, cryptography-based solutions and their detailed comparisons in addition to considerations on issues (i.e., implementation flaws, users and usability, physical security of the devices and trade offs), taxonomy of D/DoS attacks for IoT, analysis of the studies which analyze the effects of the attacks based on the attacks and findings, examination of mitigation systems and protocol security solutions with respect to mitigation mechanisms and targeted attacks, categorization of D/DoS attacks according to detection architecture, detection technique, detection resources as well as targeted attacks and implementation environments.

Although we can think that cryptography will be enough for us, various issues open our networks to D/DoS attacks. D/DoS attacks are clearly threats not only for availability but also for reliability of future Internet of Services. There are various attacks and literature has several studies to secure the IoT networks against these attacks. When we consider the efforts, we cannot say that IoT security is over now. Clearly, there is still a lot to research and consider.

Although majority of studies examined in this work target 6LoWPAN networks, security of emerging communication technologies such as LoRaWAN, NB-IoT, Thread and many others needs attention of researchers.

Based on our analysis, we can say that a plethora of research exists for routing layer D/DoS attacks, whereas we can not see studies targeting the application layer of IoT. Therefore, security of the application layer considering the attacks and use-cases needs research. In addition to this, most of the studies do not focus on a broad range of attacks, but only a few. There is a need for proposals which are capable of targeting more attacks for IoT security research.

Only a few papers consider IoT to be used as an attacking tool for D/DoS attacks by malicious entities. However, the predicted number of devices in IoT networks is in the order of billions and IoT applications will be weaved into the fabric of our daily lives. It will be very easy for attackers to target. Therefore, there is a serious need for studies which address this issue.

Nevertheless, while researchers will focus into the mentioned issues as future research, they will face with several challenges including resource limitations, heterogeneity of devices and applications, usability and security awareness, management and cost.

Acknowledgments. This study was supported by COST Action IC1304 with STSM reference ECOST-STSM-IC1304-010217-081547. It was also supported by 2211C - Domestic Doctoral Scholarship Program Intended for Priority Areas, No. 1649B031503218 of the Scientific and Technological Research Council of Turkey (TUBITAK).

References

1. IEEE Standard for Local and Metropolitan Area Networks - Part 15.4: Low Rate Wireless Personal Area Networks. IEEE Std. 802.15.4-2011 (2011)
2. DDoS on Dyn Impacts Twitter, Spotify, Reddit (2016). <https://krebsonsecurity.com/2016/10/ddos-on-dyn-impacts-twitter-spotify-reddit/>. Accessed 7 Dec 2016

3. Digital Attack Map Top daily DDoS attacks worldwide (2018). <http://www.digitalattackmap.com/>. Accessed 19 Jan 2018
4. Acar, Y., Backes, M., Fahl, S., Kim, D., Mazurek, M.L., Stransky, C.: How Internet resources might be helping you develop faster but less securely. *IEEE Secur. Priv.* **15**(2), 50–60 (2017). <https://doi.org/10.1109/MSP.2017.24>
5. Adat, V., Gupta, B.B.: Security in Internet of Things: issues, challenges, taxonomy, and architecture. *Telecommun. Syst.* (2017). <https://doi.org/10.1007/s11235-017-0345-9>
6. Amaral, J.P., Oliveira, L.M., Rodrigues, J.J.P.C., Han, G., Shu, L.: Policy and network-based intrusion detection system for IPv6-enabled wireless sensor networks. In: 2014 IEEE International Conference on Communications (ICC), pp. 1796–1801 (2014)
7. Amin, Y.M., Abdel-Hamid, A.T.: A comprehensive taxonomy and analysis of IEEE 802.15.4 attacks. *J. Electr. Comput. Eng.*, 1–12 (2016). <https://doi.org/10.1155/2016/7165952>
8. Arce, I., Clark-Fisher, K., Daswani, N., DelGrosso, J., Dhillon, D., Kern, C., Kohno, T., Landwehr, C., McGraw, G., Schoenfeld, B., et al.: Avoiding the top 10 software security design flaws. Technical report, IEEE Computer Societys Center for Secure Design (CSD) (2014)
9. Aris, A., Oktug, S.F.: Poster: state of the art ids design for IoT. In: International Conference on Embedded Wireless Systems and Networks (EWSN 2017) (2017)
10. Aris, A., Oktug, S.F., Yalcin, S.B.O.: Internet-of-Things security: denial of service attacks. In: 2015 23th Signal Processing and Communications Applications Conference (SIU), pp. 903–906 (2015). <https://doi.org/10.1109/SIU.2015.7129976>
11. Aris, A., Oktug, S.F., Yalcin, S.B.O.: RPL version number attacks: in-depth study. In: NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pp. 776–779 (2016). <https://doi.org/10.1109/NOMS.2016.7502897>
12. Bormann, C., Hoffman, P.: Concise Binary Object Representation (CBOR). RFC 7049 (Proposed Standard) (2013). <https://doi.org/10.17487/RFC7049>. <https://www.rfc-editor.org/rfc/rfc7049.txt>
13. Brandt, A., Buron, J., Porcu, G.: Home Automation Routing Requirements in Low-Power and Lossy Networks. RFC 5826 (Informational) (2010). <http://www.ietf.org/rfc/rfc5826.txt>
14. Cervantes, C., Poplade, D., Nogueira, M., Santos, A.: Detection of sinkhole attacks for supporting secure routing on 6LoWPAN for Internet of Things. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 606–611 (2015)
15. Cho, E.J., Kim, J.H., Hong, C.S.: Attack model and detection scheme for botnet on 6LoWPAN. In: Hong, C.S., Tonouchi, T., Ma, Y., Chao, C.-S. (eds.) APNOMS 2009. LNCS, vol. 5787, pp. 515–518. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04492-2_66
16. Dvir, A., Holczer, T., Buttyan, L.: VeRA - version number and rank authentication in RPL. In: 2011 IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS), pp. 709–714 (2011)
17. Evans, D.: The Internet of Things how the next evolution of the internet is changing everything (2011). http://www.cisco.com/web/about/ac79/docs/innov/IoT_IBSG_0411FINAL.pdf
18. Hui, J., Thubert, P.: Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC 6282 (Proposed Standard) (2011)

19. 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 2013, pp. 55–66 (2013)
20. Kasinathan, P., Costamagna, G., Khaleel, H., Pastrone, C., Spirito, M.A.: DEMO: an IDS framework for internet of things empowered by 6LoWPAN. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security, CCS 2013, pp. 1337–1340 (2013)
21. Kasinathan, P., Pastrone, C., Spirito, M., Vinkovits, M.: Denial-of-service detection in 6LoWPAN based Internet of Things. In: 2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), pp. 600–607 (2013)
22. Kent, S., Seo, K.: Security architecture for the internet protocol. RFC 4301 (Proposed Standard) (2005). <https://doi.org/10.17487/RFC4301>. <https://www.rfc-editor.org/rfc/rfc4301.txt>. Updated by RFCs 6040, 7619
23. Kettimuthu, R., Muthukrishnan, S.: Is bluetooth suitable for large-scale sensor networks? In: ICWN, pp. 448–454. Citeseer (2005)
24. Khan, R., Khan, S.U., Zaheer, R., Khan, S.: Future internet: the Internet of Things architecture, possible applications and key challenges. In: 2012 10th International Conference on Frontiers of Information Technology, pp. 257–260 (2012). <https://doi.org/10.1109/FIT.2012.53>
25. Krco, S.: Bluetooth based wireless sensor networks—implementation issues and solutions (2002). <http://www.telfor.org.yu/radovi/4019.pdf>
26. Kurose, J.F., Ross, K.W.: Computer Networking: A Top-Down Approach, 6th edn. Pearson (2012)
27. Le, A., Loo, J., Chai, K.K., Aiash, M.: A specification-based IDS for detecting attacks on RPL-based network topology. Information **7**(2) (2016). <https://doi.org/10.3390/info7020025>. <http://www.mdpi.com/2078-2489/7/2/25>
28. 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. **13**(10), 3685–3692 (2013). <https://doi.org/10.1109/JSEN.2013.2266399>
29. Le, A., Loo, J., Luo, Y., Lasebae, A.: Specification-based IDS for securing RPL from topology attacks. In: 2011 IFIP Wireless Days (WD), pp. 1–3 (2011). <https://doi.org/10.1109/WD.2011.6098218>
30. Levis, P., Clausen, T., Hui, J., Gnawali, O., Ko, J.: The Trickle algorithm. RFC 6206 (Proposed Standard) (2011). <https://doi.org/10.17487/RFC6206>. <https://www.rfc-editor.org/rfc/rfc6206.txt>
31. Martocci, J., Mil, P.D., Riou, N., Vermeylen, W.: Building automation routing requirements in low-power and lossy networks. RFC 5867 (Informational) (2010). <http://www.ietf.org/rfc/rfc5867.txt>
32. Mayzaud, A., Badonnel, R., Chrisment, I.: Detecting version number attacks using a distributed monitoring architecture. In: Proceedings of IEEE/IFIP/In Association with ACM SIGCOMM International Conference on Network and Service Management (CNSM 2016), pp. 127–135 (2016)
33. Mayzaud, A., Sehgal, A., Badonnel, R., Chrisment, I., Schwluder, J.: Mitigation of topological inconsistency attacks in RPL-based low-power lossy networks. Int. J. Netw. Manag. **25**(5), 320–339 (2015). <https://doi.org/10.1002/nem.1898>
34. Mayzaud, A., Sehgal, A., Badonnel, R., Chrisment, I., Schwluder, J.: Using the RPL protocol for supporting passive monitoring in the Internet of Things. In:

- NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pp. 366–374 (2016). <https://doi.org/10.1109/NOMS.2016.7502833>
35. Mayzaud, A., Sehgal, A., Badonnel, R., Chrisment, I., Schönwälder, J.: A study of RPL DODAG version attacks. In: Sperotto, A., Doyen, G., Latré, S., Charalambides, M., Stiller, B. (eds.) AIMS 2014. LNCS, vol. 8508, pp. 92–104. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-43862-6_12
 36. Misra, S., Krishna, P.V., Agarwal, H., Saxena, A., Obaidat, M.S.: A learning automata based solution for preventing distributed denial of service in Internet of Things. In: Proceedings of the 2011 International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, ITHINGSCSPCOM 2011, pp. 114–122 (2011)
 37. Palattella, M., Accettura, N., Vilajosana, X., Watteyne, T., Grieco, L., Boggia, G., Dohler, M.: Standardized protocol stack for the internet of (Important) things. *IEEE Commun. Surv. Tutor.* **15**(3), 1389–1406 (2013)
 38. Pister, K., Thubert, P., Dwars, S., Phinney, T.: Industrial routing requirements in low-power and lossy networks. RFC 5673 (Informational) (2009). <http://www.ietf.org/rfc/rfc5673.txt>
 39. Pongle, P., Chavan, G.: Article: real time intrusion and wormhole attack detection in Internet of Things. *Int. J. Comput. Appl.* **121**(9), 1–9 (2015)
 40. Raza, S., Duquenooy, S., Chung, T., Yazar, D., Voigt, T., Roedig, U.: Securing communication in 6LoWPAN with compressed IPsec. In: 2011 International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), pp. 1–8 (2011). <https://doi.org/10.1109/DCOSS.2011.5982177>
 41. Raza, S., Wallgren, L., Voigt, T.: SVELTE: real-time intrusion detection in the Internet of Things. *Ad Hoc Netw.* **11**(8), 2661–2674 (2013)
 42. Razzak, F.: Spamming the Internet of Things: a possibility and its probable solution. *Proc. Comput. Sci.* **10**, 658–665 (2012). <https://doi.org/10.1016/j.procs.2012.06.084>. <http://www.sciencedirect.com/science/article/pii/S1877050912004413>
 43. Renuka Venkata Ramani, C.: Two way Firewall for Internet of Things. Master's thesis. KTH, School of Electrical Engineering (EES) (2016)
 44. Rescorla, E., Modadugu, N.: Datagram transport layer security. RFC 4347 (Proposed Standard) (2006). <https://doi.org/10.17487/RFC4347>. <https://www.rfc-editor.org/rfc/rfc4347.txt>. Obsoleted by RFC 6347, updated by RFCs 5746, 7507
 45. Saeed, A., Ahmadinia, A., Javed, A., Larjani, H.: Intelligent intrusion detection in low-power IoTs. *ACM Trans. Internet Technol.* **16**(4), 27:1–27:25 (2016). <https://doi.org/10.1145/2990499>. <http://doi.acm.org/10.1145/2990499>
 46. Samaila, M.G., Neto, M., Fernandes, D.A.B., Freire, M.M., Inácio, P.R.M.: Security challenges of the Internet of Things. In: Batalla, J.M., Mastorakis, G., Mavromoustakis, C.X., Pallis, E. (eds.) Beyond the Internet of Things. IT, pp. 53–82. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-50758-3_3
 47. Schneier, B.: Stop trying to fix the user. *IEEE Secur. Priv.* **14**(5), 96 (2016). <https://doi.org/10.1109/MSP.2016.101>
 48. Sedjelmaci, H., Senouci, S.M., Al-Bahri, M.: A lightweight anomaly detection technique for low-resource IoT devices: a game-theoretic methodology. In: 2016 IEEE International Conference on Communications, ICC 2016 (2016). <https://doi.org/10.1109/ICC.2016.7510811>
 49. Sehgal, A., Mayzaud, A., Badonnel, R., Chrisment, I., Schönwälder, J.: Addressing DODAG inconsistency attacks in RPL networks. In: Global Information Infrastructure and Networking Symposium (GIIS 2014), pp. 1–8 (2014)
 50. Shelby, Z., Hartke, K., Bormann, C.: The Constrained Application Protocol (CoAP). RFC 7252 (Proposed Standard) (2014)

51. Sokullu, R., Dagdeviren, O., Korkmaz, I.: On the IEEE 802.15.4 MAC Layer Attacks: GTS Attack. In: Second International Conference on Sensor Technologies and Applications, SENSORCOMM 2008, pp. 673–678 (2008)
52. Tsao, T., Alexander, R., Dohler, M., Daza, V., Lozano, A., Richardson, M.: A security threat analysis for the routing protocol for low-power and Lossy Networks (RPLs). RFC 7416 (Informational) (2015). <http://www.ietf.org/rfc/rfc7416.txt>
53. UC Berkeley Center for Long-Term Cybersecurity: Cybersecurity Futures 2020. Technical report (2016)
54. Wallgren, L., Raza, S., Voigt, T.: Routing attacks and countermeasures in the RPL-based internet of things. *Int. J. Distrib. Sens. Netw.* **2013**, 11 (2013)
55. Weekly, K., Pister, K.: Evaluating sinkhole defense techniques in RPL networks. In: 2012 20th IEEE International Conference on Network Protocols (ICNP), pp. 1–6 (2012)
56. Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, J., Alexander, R.: RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550 (Proposed Standard) (2012)
57. Wu, M., Lu, T.J., Ling, F.Y., Sun, J., Du, H.Y.: Research on the architecture of Internet of Things. In: 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), vol. 5, pp. V5–484–V5–487 (2010). <https://doi.org/10.1109/ICACTE.2010.5579493>
58. Yang, Y., Wu, L., Yin, G., Li, L., Zhao, H.: A survey on security and privacy issues in Internet-of-Things. *IEEE Internet of Things J.* **4**(5), 1250–1258 (2017). <https://doi.org/10.1109/JIOT.2017.2694844>
59. Yang, Z., Yue, Y., Yang, Y., Peng, Y., Wang, X., Liu, W.: Study and application on the architecture and key technologies for IoT. In: 2011 International Conference on Multimedia Technology, pp. 747–751 (2011). <https://doi.org/10.1109/ICMT.2011.6002149>
60. Zarpelo, B.B., Miani, R.S., Kawakani, C.T., de Alvarenga, S.C.: A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* **84**, 25 – 37(2017). <https://doi.org/10.1016/j.jnca.2017.02.009>. <http://www.sciencedirect.com/science/article/pii/S1084804517300802>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter’s Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

