Insecure to the Touch: Attacking ZigBee 3.0 via Touchlink Commissioning

Conference Paper · July 2017

DOI: 10.1145/3098243.3098254

CITATIONS
10

READS
912

5 authors, including:

Philipp Morgner
Friedrich-Alexander-University of Erlangen-Nürnberg
10 PUBLICATIONS **26** CITATIONS

SEE PROFILE

Insecure to the Touch: Attacking ZigBee 3.0 via Touchlink Commissioning

Philipp Morgner
Friedrich-Alexander-Universität Erlangen-Nürnberg
philipp.morgner@fau.de

Stephan Mattejat
Friedrich-Alexander-Universität Erlangen-Nürnberg
stephan.mattejat@fau.de

Zinaida Benenson
Friedrich-Alexander-Universität Erlangen-Nürnberg
zinaida.benenson@fau.de

Christian Müller
University of Mannheim
christian.mueller@uni-mannheim.de

Frederik Armknecht
University of Mannheim
armknecht@uni-mannheim.de

ABSTRACT

Hundred millions of Internet of Things devices implement ZigBee, a low-power mesh network standard, and the number is expected to be growing. To facilitate an easy integration of new devices into a ZigBee network, touchlink commissioning was developed. It was adopted in the latest specifications, ZigBee 3.0, which were released to the public in December 2016, as one of two commissioning options for ZigBee devices. ZigBee 3.0 products can be used in various applications, also including security-critical products such as door locks and intruder alarm systems. The aim of this work is to warn about a further adoption of this commissioning mode. We analyze the security of touchlink commissioning procedure and present novel attacks that make direct use of standard's features, showing that this commissioning procedure is insecure by design. We release an open-source penetration testing framework to evaluate the practical implications of these vulnerabilities. Evaluating our tools on popular ZigBee-certified products, we demonstrate that a passive eavesdropper can extract key material from a distance of 130 meters. Furthermore, an active attacker is able to take-over devices from distances of 190 meters. Our analysis concludes that even a single touchlink-enabled device is sufficient to compromise the security of a ZigBee 3.0 network, and therefore, touchlink commissioning should not be supported in any future ZigBee products.

CCS CONCEPTS

•**Security and privacy** →**Mobile and wireless security;** *Distributed systems security;*

KEYWORDS

ZigBee, Internet of Things, IoT, Security, Touchlink, Commissioning

1 INTRODUCTION

ZigBee is a popular standard for wireless low-power communication in the Internet of Things (IoT), especially in the domain of smart home networks. The ZigBee Alliance, a non-profit organization of more than 400 member companies maintaining the ZigBee specifications, lists more than 1,300 certified products [24], and claims to have the largest base of installed IoT devices worldwide [21] with more than a hundred million devices [23].

In December 2016, the latest ZigBee specifications, denoted as ZigBee 3.0, were released to the public. These specifications define function clusters for several smart home applications including security-critical applications such as door locks, window shades, and intruder alarm systems. To prevent the manipulation and unauthorized control of these applications, appropriate security measures are crucial for ZigBee networks. One of the most critical parts in the security design is commissioning, which is the procedure of either starting a new network or integrating a new node into an existing network. During the process of joining of a new node to an existing network, this node needs to be equipped with the network key in a secure manner, which is a challenging task for heterogeneous IoT networks that interconnect products of multiple manufacturers.

ZigBee 3.0 provides two different commissioning procedures to accomplish this task: EZ-Mode commissioning and touchlink commissioning. In this paper, we focus on touchlink commissioning, which was originally developed to easily integrate devices in connected lighting systems that follow the (legacy) ZigBee Light Link standard. The basic idea of touchlink commissioning is to facilitate close physical proximity instead of cryptographic authentication for joining new nodes to a network. The ZigBee 3.0 specifications inherited the touchlink commissioning procedure as a commissioning option for ZigBee 3.0 products without giving guidelines whether an application is suitable for touchlink commissioning.

Our contribution is twofold. As first contribution, we provide a security analysis of the ZigBee touchlink commissioning procedure, which has not been part of a comprehensive security analysis before, to the best of our knowledge. During our investigations, we analyzed the specifications and learned that the touchlink communication relies on inter-PAN frames, which are neither secured nor authenticated. Furthermore, the transport of the network key to a joining device is protected solely by a global master key, the touchlink preconfigured link key. This key is distributed to manufacturers of touchlink-enabled products under a non-disclosure agreement (NDA) but was leaked in March 2015 and cannot be renewed due to the backward compatibility demands towards legacy ZigBee Light Link products. In addition, we learned from the specifications that

WiSec '17, Boston, MA, USA

DOI: 10.1145/3098243.3098254

the distance check between the joining node and the initiator is based on a simple signal strength threshold.

As second contribution, we developed and evaluated a real-world attack system to eavesdrop and inject packets in the communication of ZigBee networks. In this context, we release the open-source penetration testing framework *Z3sec* that is able to create arbitrary touchlink commands and provides an interface to control ZigBee-certified devices once the network key is known. We evaluated our penetration testing framework on popular ZigBee-certified and touchlink-enabled products of four different manufacturers. In our evaluation, we demonstrated the extraction of the current network key from a distance of 130 meters through passively eavesdropping on a touchlink commissioning procedure. In the domain of active attacks, we were able to permanently disconnect nodes from the legitimate network, or to reset them to factory-new. We can also trigger the so-called identify action, e.g., causing light bulbs to blink, for several hours. Furthermore, we demonstrated that we can remove nodes from their legitimate networks and join them to the attacker's network. In our evaluation, we were able to perform such an active attack from a distance between 15 and 190 meters depending on the tested products. Due to limitations of the experimental setup, longer distances might be possible.

In conclusion, our evaluation shows that the support of touchlink commissioning is sufficient to compromise the security of ZigBee 3.0 applications. In our threat scenarios, we outline that already a single touchlink-enabled device allows attackers to take control over arbitrary devices in the ZigBee network, also including security-critical applications.

The paper is structured as follows. We introduce the ZigBee 3.0 standard in Section 2. In Section 3, we present the threat model for the security analysis. In Section 4, we perform the security analysis of the touchlink commissioning procedure, which contains the description of novel attacks as well as their practical evaluation. In Section 5, we describe the disclosure to the manufacturers and their responses. We discuss the results of our evaluation, consequences, and mitigation in Section 6. We present related work in Section 7, and this paper concludes in Section 8.

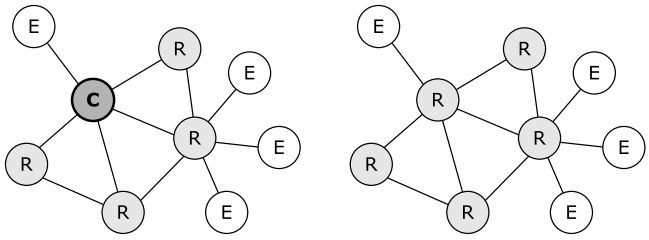
2 ZIGBEE

ZigBee is a wireless low-power standard that connects embedded technologies in wireless personal area networks (WPANs). Compared to Wi-Fi, ZigBee-certified devices send smaller packets and consume far less energy, while ZigBee has a larger wireless range than Bluetooth.

The ZigBee specifications are maintained by the ZigBee Alliance, a global non-profit organization that comprises over 400 member companies. The ZigBee Alliance defines the network, security, and application layers and supervises the conformance and interoperability of ZigBee-certified products. The *ZigBee 3.0* specifications, which were ratified in December 2015 and released to the public in December 2016, replace the *ZigBee Pro* specifications [25] from 2012. The main difference between ZigBee 3.0 and ZigBee Pro is that the ZigBee Pro specifications facilitated several *application profiles* comprising customized sets of features and protocols for specific application areas. Examples of such profiles are ZigBee Home Automation [19] for applications in residential environments, ZigBee Smart Energy [18] for smart metering, or ZigBee Light Link [17] for connected lighting systems. The fragmentation in the ZigBee Pro standard resulted in interoperability problems between ZigBee-certified products of different profiles such that the ZigBee Alliance decided to merge these profiles into one standard, which is ZigBee 3.0. An exception is the Smart Energy profile, which remains independent due to special requirements of smart metering applications. The ZigBee 3.0 standard is defined in multiple specification documents, of which the ZigBee 3.0 Base Device Behavior specification [20] and the ZigBee 3.0 Cluster Library specification [22] are publicly available.

2.1 System Model

The ZigBee 3.0 specifications describe three logical types of nodes: *coordinator*, *router*, and *end device*. Each node can comprise one or more devices, and at any point in time is designated to only one of the logical types. Coordinators and routers are usually devices that have permanent power supply, in contrast to end devices, which are usually battery-powered.



(a) Centralized security **(b) Distributed security**

Figure 1: Security network models. Notation: C = coordinator, R = router, E = end device.

Each ZigBee 3.0 network is either a distributed or a centralized security network. As illustrated in Figure 1a, a *centralized security network* is managed by a coordinator that includes the trust center. This coordinator authenticates new nodes and joins them to the network. In contrast, a *distributed security network* is formed by a router and has no coordinator as shown in Figure 1b. A new node is authenticated and joined to the network by an arbitrary router, which becomes its parent node.

2.2 Security

The ZigBee 3.0 stack sits on top of the physical layer and medium access control (MAC) layer defined in the IEEE 802.15.4 specifications [7]. Security measures in ZigBee applications are only applied to the network and application layer. Although the MAC layer of the IEEE 802.15.4 standard specifies multiple encryption and authentication mechanisms, these mechanisms are not used in ZigBee applications. ZigBee-certified devices facilitate the AES-CCM* authenticated encryption scheme[1] with an 128-bit *network key*. This network key is shared between all devices of a network and used to secure the communication.

[1] Compared to AES-CCM (without asterisk), this specific mode allows also encryption-only or integrity-only variants.

2.3 Commissioning

Commissioning is the process in which either a new ZigBee network is started, or a new node is joined to an existing ZigBee network. The ZigBee 3.0 standard specifies two commissioning procedures: *EZ-Mode commissioning* and *touchlink commissioning*. While the support of EZ-Mode commissioning is mandatory for each ZigBee 3.0 device, manufacturers can decide whether touchlink commissioning is enabled in their products. The specifications do not provide any guideline if touchlink commissioning is appropriate for a certain application. In addition, touchlink is supported by all legacy devices that follow the ZigBee Light Link specifications.

2.3.1 EZ-Mode Commissioning. The EZ-Mode must be invoked with a user action, e.g., through pushing a button on the device. After this mode is activated, the node is put into EZ-Mode for a time frame of 3 minutes, which can be extended through further user actions.

In EZ-Mode, a node that is not joined to a network, scans for open networks in its wireless range. In case the node finds a suitable network, it attempts to join this network using MAC association as specified in IEEE 802.15.4. If the network allows the node to join, the node waits to become authenticated and receives the network key. In a centralized security network, the network key is encrypted using either the publicly known *default global Trust Center link key*, which is provided in the ZigBee specifications, or via a preconfigured link key that is derived from an *install code*, which is a unique code printed on the node in a manufacturer-specific fashion. In a distributed security network, the network key is transmitted after being encrypted using the NDA-protected *distributed security global link key*.

2.3.2 Touchlink Commissioning. The touchlink commissioning procedure was first introduced in the ZigBee Light Link standard, and later adopted by the ZigBee 3.0 specifications. Touchlink commissioning is patented by Philips [8] and was specifically designed to make connected lighting systems easy to deploy and use for consumers. Compared to other commissioning options, touchlink commissioning provides an extended functionality that goes beyond the plain joining of devices. The objective was to enable use cases in which commissioning is performed between a bulb and a low-function device, e.g., a remote control. For such scenarios, touchlink commissioning offers the possibility to manage network features, such as reset to factory-new or channel switch, with the so-called *touchlink commands*.

Figure 2 describes the commissioning protocol for joining a device, denoted as target or end device, to an existing ZigBee network. The initiator is usually a remote control or a bridge device that is connected to the Internet. First, the initiator starts the device scan procedure by sending *scan requests* on specific channels as defined in the specifications. These scan requests include a randomly generated transaction identifier. The target replies with a *scan response* containing the same transaction identifier, a random response identifier, and further information. The device scan may yield multiple potential nodes from which the user can select one for the next steps. The user has the option to send an *identify request* to a device, upon which the target performs a pre-defined identify action, e.g., a bulb flashes for a few seconds. An *identify request* contains the corresponding transaction identifier as well as the duration of the identify action.

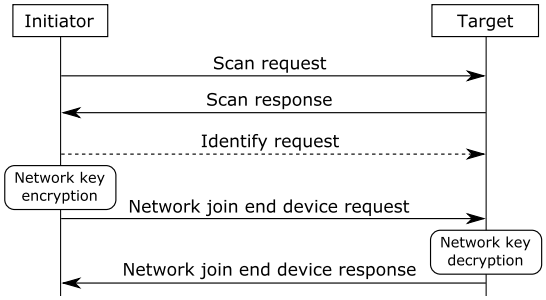


Figure 2: Touchlink commissioning protocol.

To join a new node to a network, the initiator encrypts the current network key with the *touchlink preconfigured link key*, builds a *network join end device request* containing the encrypted network key, transaction identifier as well as further network information, and then sends this command frame to the selected node. On receiving the message, the joining node decrypts the network key using the touchlink preconfigured link key, and replies with a *network join end device response* indicating success or failure.

3 THREAT MODEL

3.1 Security Goals and Attacker Model

The ZigBee specifications describe security assumptions [25, p. 426], such as safekeeping of key material and proper implementation of security protocols, but do not define security goals [1]. For this reason, we define following *security goals* that apply to ZigBee networks:

- Confidentiality: Only legitimate entities are allowed to access data and commands sent within the network.
- Integrity: Data and commands sent within the network are not tampered with.
- Authenticity: The receiver is able to reject commands and data sent by illegitimate entities.
- Availability: The functionalities and data of the devices in the network are continuously available to all legitimate entities.

The threat model is determined as follows: The user of the ZigBee network is trusted and honest, and installs the network as required by the manufacturer. The online account credentials for the remote access to the ZigBee network are not disclosed. The nodes of the network are certified by the ZigBee Alliance and follow the protocols described in the ZigBee Light Link or the ZigBee 3.0 specifications. Therefore, all touchlink-enabled ZigBee devices are equipped with the touchlink preconfigured link key.

The goal of the attacker is to violate any of the security goals mentioned above. We assume that attackers have neither physical access to the ZigBee devices nor to the local area network (LAN) or wireless LAN (WLAN) to which a ZigBee device might be connected. The only capability of the attacker is to eavesdrop and inject packets in the wireless communication of at least one touchlink-enabled node of the targeted ZigBee network.

Thus, the attacker controls an IEEE 802.15.4 radio transceiver, which is present within the wireless range of the targeted node.

Note that the radio transceiver can be mounted on a drone such that no close physical proximity of the person that performs the attack is required. This potential scenario was demonstrated in [12].

3.2 Threat Scenarios

There exists a wide variety of ZigBee-certified devices in the smart home domain, from sensors like smoke detectors and hygrometers, controllers for building services to appliances like washing machines. The ZigBee 3.0 specifications define also function clusters for security-critical applications, such as front door locks, window shade controllers, and intruder alarm systems. Thus, the attack surface, which we evaluate on touchlink-enabled connected lighting systems, potentially affects all future ZigBee-certified products that are compliant to either the ZigBee Light Link or ZigBee 3.0 specifications.

Security-critical ZigBee devices can be attacked in two ways: either directly, in case the targeted ZigBee device supports touchlink commissioning, or indirectly, in case there exists at least one touchlink-enabled device in a ZigBee network that contains security-critical devices. In the following, we outline three threat scenarios.

Scenario #1. We assume an application that authorizes door access to a restricted area. The network is organized as centralized security network and communicates based on ZigBee 3.0 standard. One of the nodes of the network is a ZigBee-certified door lock that implements touchlink commissioning. As we show in Section 4.3.3, the attacker can reset arbitrary touchlink-enabled nodes to factory-new without knowing any cryptographic secrets. The attacker applies this reset attack to the door lock, which puts the lock in factory-new state and most probably clears the way for the attacker.

Scenario #2. We assume a smart home network comprising various applications including an intruder alarm system. This network is organized as a ZigBee 3.0 distributed security network. We further assume that an attacker can eavesdrop the touchlink commissioning procedure of an arbitrary touchlink-enabled device, which is joined to this network, e.g., a new light bulb. From this captured communication, the attacker extracts the network key as described in Section 4.4.2. As a consequence, the attacker can decrypt all further network communication as well as inject commands into the network. Hence, the attacker can reset the intruder alarm system to factory-new through sending spoofed network leave commands, and break into the house without triggering the alarm.

Scenario #3. A household intended for elderly living deploys a distributed security network consisting of a large number of touchlink-enabled devices. In Sections 4.3.2 and 4.3.4, we present attacks that allow to permanently disconnect touchlink-enabled devices from their legitimate network and to trigger a manufacturer-specific identify action for a duration up to 18 hours. For example, in case of light bulbs, this action is blinking but for other devices this can also be making sounds or moving. The attack proceeds as follows: first, the attacker permanently disconnects all touchlink-enabled devices, and then makes them blink, beep, or move for several hours. On payment, the attacker promises to stop the attacks and to reconnect the devices to the legitimate network. The residents can decide whether they want to manually reset and recommission each device or to pay the demanded amount of money. Since the recovery of ZigBee devices from the permanent disconnect attack can be an extremely cumbersome task that requires dexterity and precise timing, as described in Section 4.6, residents might prefer to pay the ransom.

4 SECURITY ANALYSIS OF THE TOUCHLINK COMMISSIONING PROCEDURE

We divide our attacks into two types: passive and active attacks. Passive attacks only eavesdrop on the wireless communication of nodes in the targeted network, while active attacks require the attacker to interact with the targeted node via wireless communication. In addition, we also categorize our attacks according to the goal of the attacker: In the first category, we describe denial-of-service (DoS) attacks that exploit security weaknesses in the concept of so-called inter-PAN frames. These attacks require no knowledge of any cryptographic material. In the second category, we show attacks that allow the attacker to control devices in the network. These attacks require knowledge of the touchlink preconfigured link key that has been leaked in March 2015. In Table 1, we provide an overview of all attacks that are described in this section.

Table 1: Overview of attacks.

	Type of Attack		Attacker Goals	
Attack	**Active**	**Passive**	**DoS**	**Control**
Identify Action	●	○	●	○
Reset to Factory-New	●	○	●	○
Permanent Disconnect	●	○	●	○
Hijack	●	○	●	●
Network Key Extraction	○	●	○	●

The attacks in this section outline the procedures to compromise a single touchlink-enabled device. All attacks can be easily extended to target multiple devices at the same time by running the attack procedures for different target devices simultaneously.

4.1 Penetration Testing Framework Z3sec

For our research, we developed the penetration testing framework *Z3sec* in Python to evaluate the security of ZigBee 3.0 devices. These tools and their documentation are available as open-source software on GitHub[2]. The Z3sec framework consists of three major components: First, a *touchlink library* to build arbitrary touchlink packets and to keep track of source addresses and sequence numbers. Second, a *crypto module* that provides the functionality to encrypt and decrypt ZigBee packets. This component also handles key transport frames, especially decrypting the encrypted network key, and vice versa. Third, the *radio interface module* enables the communication between the radio transceivers and the touchlink library.

As radio transceiver, we utilize the USRP B200 from Ettus, a software-defined radio covering the radio-frequency range between 70 MHz and 6 GHz. The USRP features an FPGA and connects to a host computer via USB 3.0. We use Scapy-radio [11] as interface to

[2]https://github.com/IoTsec/Z3sec

(a) Philips Hue (b) Osram Lightify (c) GE Link (d) IKEA Trådfri

Figure 3: The four evaluated ZigBee-certified connected lighting systems.

send and receive ZigBee packets with the USRP. Scapy-radio itself uses capabilities of GnuRadio and an IEEE 802.15.4 GnuRadio flow chart implementation [2].

In addition, we implemented a command line tool that performs the attack procedures described in the following sections, and a module that is able to send or spoof control commands to the devices once the network key is disclosed.

4.2 Testbed

We analyzed the following four systems: a Philips Hue starter set including one bridge and three white and color ambiance (LCT001) LED bulbs, which is exemplary shown in Figure 3a. Furthermore, we deployed an Osram Lightify gateway with a classic A60 tunable white LED bulb as depicted in Figure 3b. Our third system is a GE Link starter pack containing a Link hub and a Link A19 soft white LED light bulbs as shown in Figure 3c. The last system is the IKEA Trådfri system consisting of a Trådfri LED 980lm light bulb and a Trådfri remote control as depicted in Figure 3d. All these systems implement the ZigBee Light Link standard, for which touchlink commissioning is mandatory. Thus, these systems are representative for arbitrary touchlink-enabled ZigBee devices. Due to the novelty of ZigBee 3.0 specifications, only one manufacturer released ZigBee 3.0-certified products so far to the best of our knowledge (as of March 2017). These released ZigBee 3.0-certified products by Ubisys do not support touchlink commissioning. Nevertheless, the presented attacks apply to all future ZigBee 3.0 products that enable touchlink commissioning.

Before starting our evaluation, we updated the Philips Hue firmware to the then-latest version 01031131 as well as the API to version 1.12.0. We updated the Osram Lightify gateway WLAN to version 1.1.2.101 and the gateway ZigBee to version 1.2.0.67. We found no possibility to update the GE Link firmware by using the manufacturer-recommended Wink app. At the time of the evaluation, there existed neither a mobile device app for IKEA Trådfri nor a possibility to update the firmware of the Trådfri bulbs or remote control[3].

The attacker equipment comprises a laptop on which our penetration testing framework Z3sec is installed. A radio transceiver, the Ettus USRP, is connected to the laptop. We started the evaluation of each attack with the default settings, in which the lighting system works as intended and the system is not compromised.

4.3 Denial-of-Service Attacks

Our DoS attacks exploit the concept of inter-PAN frames, which are a special type of ZigBee frames that allow the communication between different personal area networks (PANs). In 2008, inter-PAN frames were introduced in the ZigBee Smart Energy application profile. In the purpose description, the ZigBee Smart Energy standard states that inter-PAN transmissions allow ZigBee devices to '*perform limited, insecure, and possibly anonymous exchange of information*' [18, p.81]. In the context of smart metering, for which the ZigBee Smart Energy standard was intended, the mandate for such a transmission mechanism is the '*market requirement to send pricing information to very low cost devices*', e.g., refrigerator magnets showing the current energy consumption or prices. The ZigBee Light Link standard adopted the inter-PAN transmission mechanism to enable the commissioning of networks with constraint devices, e.g., remote controls. In the touchlink commissioning procedure, inter-PAN frames are used to transmit touchlink commands and their responses between initiator and target device.

Since there exists no shared key material between different PANs, inter-PAN frames are neither secured nor authenticated. Hence, all attacks presented in the section are performed without requiring any knowledge of the touchlink preconfigured link key or of any other cryptographic material relating to these devices.

4.3.1 Active Device Scan. The active device scan searches for ZigBee devices in wireless range of the attacker's equipment. The active device scan is a mandatory step in preparation of any further attack.

Procedure. The attacker builds a *scan request*, then sends this inter-PAN command frame on all ZigBee channels consecutively and listens a few milliseconds on each channel for *scan responses*. Through the reception of *scan responses*, the attacker learns about all ZigBee devices that are also listening on this channel. ZigBee uses 16 channels in the 2.4 GHz ISM band: channel 11 to 26, while channel 1 to 10 are located in other ISM bands. The ZigBee 3.0 specifications define four primary channels on which devices are listening for touchlink scans: 11, 15, 20, and 25. These channels are used for commissioning and normal operations, while all remaining channels can be used as backup.

Evaluation. The active device scan works with all four lighting systems. In general, all light bulbs responded to the *scan request*, while also the Osram Lightify gateway answers each time. The GE Link hub does not respond to *scan requests*, and the Philips Hue bridge only replies if the button on the hub was pushed within the last 30 seconds.

ACK Spoofing. All following attacks start with an active device scan and then send further inter-PAN command frames. When target devices in Osram Lightify or GE Link implementations receive a *scan request*, they answer with the *scan response* and then wait for a MAC-layer acknowledgment (ACK). If they do not receive an ACK within a specified time frame of 864 microseconds, they drop further communication. Z3sec is too slow in sending

[3]IKEA released a Trådfri gateway and a Trådfri mobile device app in April 2017.

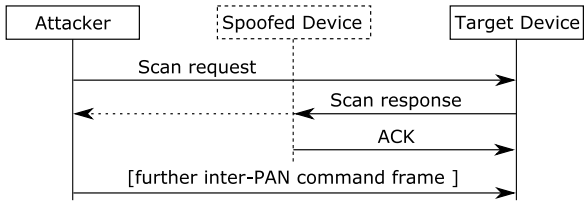


Figure 4: Acknowledgment spoofing: the attacker spoofs a third device to send an ACK to the target device timely.

this ACK since the processing of the received ZigBee frames is not performed on the hardware platform but in software, and therefore delayed for a few milliseconds. However, we can impersonate an existing ZigBee device, referred to as spoofed device in Figure 4, by setting the extended source address of the *scan request* to the extended address of the spoofed device. As a result, the spoofed device sends an ACK upon reception of a *scan response*, even if this device had never sent any *scan request*. This is an inherent feature of the IEEE 802.15.4 MAC layer, and we can leverage this mechanism for "providing" ACKs timely. On the contrary, Philips Hue and IKEA Trådfri devices do not require any ACKs to the *scan responses*, making our attacks easier to implement.

4.3.2 Identify Action Attack. The touchlink commissioning procedure provides the possibility to request a ZigBee device to identify itself via a pre-defined identify action, e.g., flashing, dimming, or beeping. Originally, the identify procedure is intended to give the user a possibility to select and identify a certain node, which should be added to the network but the identify action also can be abused by attackers.

Procedure. After an active device scan, the attacker can send an *identify request* to the targeted ZigBee device. The *identify request* contains the transaction identifier and identify duration. The identify duration can be at maximum 0xFFFE, which converts to a time duration of 18 hours 12 minutes 14 seconds. If the identify duration is set to 0, a previously started identify procedure is aborted before the specified duration elapsed. Setting the identify duration to 0xFFFF requests the device to perform the identify procedure for a device-specific default period of time, usually a few seconds. At the reception of the *identify request*, the targeted ZigBee device starts its identify action for the defined period of time.

Evaluation. All lighting systems are vulnerable to the identify action attack. During the blinking of the lights, the users can neither turn off nor control the light bulb using the apps provided by the manufacturers. The only way to shut down the lights is to physically disconnect the bulb from the power supply. An exception are IKEA Trådfri bulbs, which dim their lights up and down instead of flashing. Also, the identify action can be immediately aborted by pressing an arbitrary button on the remote control.

The attacker can abort the attack anytime by sending another *identify request* with the field duration set to zero. The maximum duration of blinking that can be triggered with a single *identify request* is shown in Table 2. We assume that the duration depends on the manufacturer's implementation of touchlink commissioning.

After performing the identify action attack, the Philips Hue bulb and the IKEA Trådfri bulb return to the pre-attack state and color, while the Osram Lightify bulb and the GE Link bulb change to the default state and color. This attack also works if the device is turned off but supplied with power.

Table 2: Maximum duration of the identify action attack.

System	Max. duration
Philips Hue	18:12:14h
Osram Lightify	9:12:53h
GE Link	9:06:31h
IKEA Trådfri	0:01:00h

4.3.3 Reset to Factory-New Attack. In this attack, the attacker resets the configuration of a ZigBee device to the factory-new state.

Procedure. The attack is performed by sending a *reset to factory new request* inter-PAN command frame after a prior active device scan. The payload of the *reset to factory new request* only contains the transaction identifier. On the reception of a valid *reset to factory new request*, the light bulb discards the current configuration. The color and brightness of the light bulb changes to the default states.

Evaluation. Our evaluation showed that all four lighting systems are vulnerable to the reset to factory-new attack. Interestingly, we are also able to reset the Lightify gateway (at any time) as well as the Philips Hue bridge (if the button of the bridge was pushed within the last 30 seconds) to a factory-new state. After a reset to factory-new attack, the legitimate user would have to reintegrate the bulb into the legitimate network by either searching for new devices via the mobile device app or by using a remote control. This operation has to be initiated manually by the user. In the meantime, i.e., before the user initiates a recommissioning, an attacker has the chance to hijack the reset device using the classical commissioning in ZigBee Light Link or EZ-Mode commissioning in ZigBee 3.0, each in combination with the publicly known default global Trust Center link key as demonstrated in [12, 27].

4.3.4 Permanent Disconnect Attack. In the permanent disconnect attack, the user loses control over the touchlink-enabled device. This attack differs from the reset to factory-new attack in the process of recovery: after a reset to factory-new attack, the user can simply recommission the attacked bulb to the network again. In the aftermath of a permanent disconnect attack, the user needs to recover the bulb first, as described in Section 4.6, before a recommissioning to the legitimate network is possible again.

Procedure. We present two approaches to perform a permanent disconnect attack: In the first approach, we force a targeted ZigBee device to change the current channel to another channel determined by the attacker. In the second approach, we join the targeted device to a non-existent network.

A change of the wireless channel can be enforced by sending a *network update request* inter-PAN command frame. The command must include a network update identifier that is higher than the current update identifier of the targeted network, which is a counter that is incremented each time the network settings are updated. The current network update identifier can be retrieved from the *scan response* of the target device. After receiving the *network update request*, which includes the new channel, the target device switches to this channel. The legitimate network does not recognize the shift.

As a consequence, the targeted device does not receive legitimate user commands anymore.

Using a *network join end device request* inter-PAN command frame (instead of the *network update request*), an attacker can manipulate additional network settings like the PAN ID and the current network key. The attacker sets the encrypted network key field to an arbitrary 128-bit value and then sends the *network join end device request* to the targeted ZigBee device. On the reception of a *network join end device request*, the ZigBee device leaves its current network and sets the internal parameters according to the new configuration. Since the encryption of the network key is not authenticated, the device decrypts the arbitrary 128-bit value to a garbage network key, which also is not known to the attacker. The transaction is confirmed by sending a *network join end device response*.

Evaluation. In the evaluation, all four presented lighting systems are vulnerable to both permanent disconnect attacks. These attacks do neither change the color nor the state of the bulb but after performing the attack procedures, the targeted bulbs cannot be controlled by the legitimate user anymore.

4.4 Attacks to Gain Control

The authenticity and integrity of the ZigBee touchlink commissioning procedure relies on the *touchlink preconfigured link key*, also denoted as *ZLL master key* in the ZigBee Light Link specifications, that is used to encrypt the current network key before this key is transmitted to the joining device. The procedure of the network key encryption starts by expanding the transaction identifier and the response identifier from scan request and scan response, respectively, to an 128-bit string. This bit string is the input to the AES encryption function, while the touchlink preconfigured link key is used as encryption key. The resulting output is denoted as transport key. In the next step, the actual network key is encrypted with the transport key using AES encryption again.

The touchlink preconfigured link key is distributed to manufacturers of ZigBee-certified products under an NDA. However, in March 2015, the touchlink preconfigured link key was leaked on Twitter[4]. In the following, we present attack procedures in which the knowledge of the touchlink preconfigured link key is facilitated to take full control over ZigBee networks.

4.4.1 Hijack Attack. The hijack attack extends the permanent disconnect attack described in Section 4.3.4. Instead of sending arbitrary bytes as the encrypted network key, the attacker forces the ZigBee device to use an attacker-chosen network key.

Procedure. Again, the attacker builds the *network join end device request* inter-PAN command frame as described in Section 4.3.4. The attacker-chosen network key is encrypted using the leaked touchlink preconfigured link key, the transaction identifier from the *scan request* and the response identifier from the *scan response* of the targeted device. This encrypted network key is included into a *network join end device request* and sent to the targeted device. On the reception of a *network join end device request*, the device updates its internal parameters according to the received values and confirms the transaction by sending a *network join end device response*. The targeted device is now commissioned to the network of the attacker, who has full control over this device.

[4]https://twitter.com/mayazigbee

Evaluation. In the evaluation, we were able to force ZigBee devices of all four connected lighting systems to accept an attacker-chosen network key. This attack paves the way to send further application-specific commands to the targeted devices.

4.4.2 Network Key Extraction. An attacker is able to extract the current network key by eavesdropping the *scan response* and the *network join end device request*[5] of an initial touchlink commissioning. All these command frames must belong to the same transaction, i.e., contain the same transaction identifier.

Procedure. The legitimate user can be motivated to perform a touchlink commissioning procedure as a result of a prior reset to factory-new attack. Then, the user is forced to commission the node to the legitimate network again. After eavesdropping on the encrypted network key from the *network join end device request*, the network key is decrypted using the leaked touchlink preconfigured link key. The response identifier is known from the *scan response*, while the transaction identifier is included in all packets belonging to the same transaction.

Evaluation. For this attack, the legitimate user of a connected lighting system has to perform the touchlink commissioning procedure. Our investigations conclude that the Philips Hue and the IKEA Trådfri lighting systems can be targeted with this attack since only a few Philips Hue third-party apps as well as the IKEA Trådfri remote control trigger touchlink commissioning. To the best of our knowledge, there exist neither apps nor ZigBee-certified devices by Osram or GE that can initiate the touchlink commissioning procedure.

In our evaluation, we showed that all four lighting systems can be controlled once the network key is exposed. We were able to send commands to turn the bulbs on and off and to change the light color of the Philips Hue bulbs to any arbitrary color.

4.5 Evaluation of Wireless Range

In the ZigBee specification, the manufacturers are advised to limit the wireless range of touchlink commands such that only ZigBee devices in close proximity are able to perform the touchlink commissioning procedure. This limitation, denoted as *proximity check*, should be implemented in a way that the received signal strength of an initiator device must be above a certain threshold.

To get a baseline for our attacks, we measured the maximum distance to successfully perform the touchlink commissioning procedure with a Philips Hue bridge since the Osram Lightify gateway and the GE Link hub provide no possibility to trigger the touchlink commissioning procedure. The maximum distances to successfully commission a Hue bulb or a Link bulb to the Hue bridge are 1.8 meters, and 1.6 meters for a Lightify bulb. For IKEA Trådfri, the maximum distance to trigger the touchlink commissioning procedure between the remote control and a bulb are 1.5 meters.

Since the touchlink commissioning procedure is intended to require close proximity, we investigated whether the attacks work for longer distances. We set up an outdoor testbed on a sports ground, in which a line-of-sight between the USRP and the attacked bulbs was given. At the USRP, we mounted rod antennas with 8dB gain according to the manufacturer. The setup is shown in

[5]Instead of capturing the *network join end device request*, this attack can also be performed by capturing a *network join router request* or a *network start request*.

(a) Evaluated light bulbs. (b) Attacker equipment.

Figure 5: Outdoor testbed to measure the maximum distance of successfully attacking a ZigBee-certified device.

Table 3: Maximum ranges of an active attack in our evaluation.

System	Legitimate Range	Active Attack Range	
		Regular	Disclosed bug
Philips Hue	1.8m	36m	(patched)
Osram Lightify	1.6m	15m	>190m
GE Link	1.8m	28m	>190m
IKEA Trådfri	1.5m	>190m	>190m

Figure 5. We decided on an outdoor measurement with line-of-sight between the attack equipment and the target device since it is hard to generalize statements about the wireless range of these attacks through buildings. The propagation of radio waves depends on many variables, e.g., the ground plan (reflections), structure and thickness of walls, wall openings, electrical installations, as well as interference with other deployed wireless networks. Thus, we conducted outdoor measurements eliminating as many types of distortions as possible and thus providing 'ground truth' for further investigations.

4.5.1 Active attacks. In the evaluation of the active attacks, we measured the maximum distance from which we are able to trigger an identify action attack, which is described in Section 4.3.2. The identify action attack requires to pass the proximity check and to receive a response from the targeted node, and therefore, this attack is representative for active attacks. The results are shown in Table 3. The maximum distance of successfully attacking the Osram Lightify system is 15 meters, the maximum distance of the GE Link system is 28 meters and the maximum distance of the Philips Hue system is 36 meters. These distances depend on the noise of the channel as well as the orientation of the bulbs and the antennas of the USRP. We experimented with different gain and antenna settings and also with different positions and directions of the bulbs. From our measurement results, we estimate that the received signal strength of inter-PAN command frames has to be stronger than -40dBm.

We also measured the maximum distance to perform the same attack on IKEA Trådfri bulbs. In our evaluation, we were able to trigger the identify action attack from a distance of more than 190 meters. This was the maximum measurable distance due to space restrictions of the outdoor testbed. Since the range to actively attack an IKEA Trådfri bulb is much larger compared to the other systems, we assume that the proximity check is not enforced. For all these tests, the transaction identifier was chosen randomly.

Previous work [12] disclosed a bug in the ZigBee Light Link implementation of Philips Hue, in which the proximity check can be circumvented by setting the transaction identifier of the touchlink command to zero. This bug was patched by Philips in October 2016. In our evaluation, we confirmed that the same bug also affects Osram Lightify, GE Link, and IKEA Trådfri. We were able to trigger the identify action attack from the maximal measurable distance of 190 meters for each of these systems. Again, we assume that this distance can be enlarged.

4.5.2 Passive attacks. We also evaluated the ranges of passively eavesdropping on the touchlink commissioning procedure to extract the current network key. We were able to extract the network key, that was established using an IKEA Trådfri system consisting of a bulb and a remote control, from a distance of 42 meters. In addition, we extracted the network key, established between a Philips Hue bridge and a Philips Hue bulb, from a distance of 130 meters. We did not evaluate Osram Lightify as well as GE Link systems since they do not provide an interface for consumers to trigger the touchlink commissioning procedure.

4.6 Recovery

The recovery from an identify action attack or a reset to factory-new attack is easy to accomplish: for the identify action attack, the user has to disconnect the node from the power source and reconnect again. For the reset to factory-new attack, the user needs to rejoin the node to the network.

To recover from permanent disconnect and hijack attacks, which altered the configuration of the nodes, is more challenging but all evaluated lighting systems possess functions to regain control over the attacked devices. However, these procedures are not obvious at first sight. In any case, a recovery entails manual effort for the user, as we explain below.

For GE Link, Osram Lightify, and IKEA Trådfri, the only way to recover the attacked bulbs is a physical reset. The physical reset is not specified in the ZigBee Light Link standard[6], but can be achieved by powering the bulbs on and off in a certain manufacturer-specific pattern, e.g., an Osram Lightify A60 bulb must be turned on 3 seconds, off 5 seconds, and this procedure must be repeated five times. This is a cumbersome task and might not be successful at the first try. Since the physical reset mechanisms are not obvious, either a mobile device app (GE Link, Osram Lightify) or a manual (IKEA Trådfri) guides the process of performing a physical reset. The Philips Hue system lacks a physical reset, to the best of our knowledge. However, the Hue system supports an additional commissioning mechanism *manual search*, which is not specified in any ZigBee standard. *Manual search* works by entering a code that is printed on a bulb into the Hue app. The *manual search* fails if the channel of the attacked device was altered to a secondary channel.

Touchlink commissioning can be applied as an alternative recovery procedure for Philips Hue and IKEA Trådfri bulbs. However, the IKEA Trådfri remote control does not search on secondary channels, therefore, if the channel of a bulb was changed to a secondary channel, it must be reset before recommissioning. In Philips Hue, touchlink commissioning can be performed by using either the Hue

[6]The ZigBee 3.0 specifications recommend supporting a reset via local action in a manufacturer-specific fashion.

API debug tool or a third-party app. After the recommissioning with touchlink, an interesting effect can be observed: Instead of reintegrating the attacked bulb into the former network, the Philips Hue bridge detects that the network update identifier of the discovered device is higher than its own. The Hue bridge adapts to the 'latest' network settings and switches to the attacker-defined channel. Consequently, the bridge loses the connection to all other bulbs, which remain on the former channel. Afterwards, all other bulbs of the former network have to be recommissioned to the new network using touchlink. This is a time-consuming task because all devices have to be moved in close proximity (1-2 meters) to the Hue bridge in order to perform the touchlink commissioning.

5 DISCLOSURE AND RESPONSE

We reported the results of our security analysis to GE, IKEA, Philips, Osram, and the ZigBee Alliance. The manufacturers IKEA, Philips and Osram responded to our outreach[7]. In contrast, the GE Product Security Incident Response Team confirmed the reception of our notification but did not comment on our report. We discussed the results of our security analysis with representatives and members of the ZigBee Alliance and received their feedback that the analysis is accurate and complete. According to the ZigBee Alliance, the main reason for development of touchlink was to reduce the complexity of commissioning procedure for ZigBee Pro devices. Touchlink offered a low entry level for the consumers to set up connected lighting systems that are configured via remote controls. However, at the same time as the ZigBee Light Link standard was developed (2010–2012), the popularity of smartphones increased rapidly. Because of this development, the ZigBee Alliance decided to introduce a bridge device that translates TCP/IP traffic sent by smartphones into ZigBee commands, thus simplifying the handling of the classical ZigBee commissioning procedure and providing an alternative method to overcome the complexity of commissioning.

6 DISCUSSION

Summarizing the results of the security analysis, we can see that all tested ZigBee-certified products are insecure against passive and active attacks. An attacker is able to thwart the availability and can take complete control over any touchlink-enabled device. It is irrelevant whether the targeted ZigBee networks was set up using the touchlink or another commissioning procedure. Furthermore, we showed that close proximity is not required. In our evaluation, we successfully performed active attacks from a distance between 15 and 190 meters depending on the targeted product. Also, we were able to passively eavesdrop the touchlink commissioning procedure from distances between 42 and 130 meters. We assume that these distances could be further extended if the attacker uses directional antennas. We tested four different ZigBee-certified connected lighting systems that facilitate the ZigBee Light Link standard. Since the touchlink commissioning procedure in ZigBee 3.0 has not been changed compared to ZigBee Light Link, all presented attacks also apply to arbitrary ZigBee 3.0 products that enable touchlink commissioning. In summary, we state that all three threat scenarios outlined in Section 3.2 are realistic and exploit security weaknesses that exist by design. In addition, we explored that the recovery of attacked devices is quite a cumbersome task.

Usage of touchlink. Since touchlink commissioning is an optional feature in ZigBee 3.0, we recommend disabling this commissioning option in all future ZigBee products. Already a *single* touchlink-enabled device in the network can expose the network key and thus lead to insecurities of other nodes. In our communication with the ZigBee Alliance, they suggested to put the enabling of the touchlink features under application control, for example to enable touchlink only a few minutes after power-up. Although this restriction limits the vulnerability time frame, the users can be motivated by social engineering techniques to power up devices at predictable times. For example, jamming of ZigBee communication may motivate the consumers to disconnect a device from the power source and power it up again. Furthermore, the recommendation of putting touchlink commissioning under application control is not included in the specifications and so, it is not quite clear how the manufacturers should become aware of this.

Manufacturer-specific mitigation. No immediate mitigation of the attacks, presented in Section 4, is possible since the security weaknesses result from legitimate features in the specification, especially from the concept of unauthenticated inter-PAN frames. If touchlink commissioning is required, manufacturer-specific changes can be made to contain the effects of the attacks. For example, the identify action should be limited to a reasonable duration (like in the IKEA Trådfri implementation), which would decrease the impact of the identify action attack significantly.

Renewal of the touchlink preconfigured link key. Since the touchlink preconfigured link key was leaked in March 2015, the touchlink commissioning procedure is considered compromised. On the one hand, the replacement of the touchlink preconfigured link key would circumvent the attacks presented in Section 4.4, and therefore protect against take-over attacks, in which the attacker gains control over the targeted devices. On the other hand, the renewal of this key would render the integration of ZigBee Light Link-based connected lighting systems and complementary equipment into ZigBee 3.0 networks impossible. This would most likely lead to public resentment, as the following incident illustrates. In December 2015, an update of the Hue app by Philips locked out light bulbs of other vendors if these vendors did not participate in the 'Friends of Hue' certification program. The public outcry made Philips revert this decision after a few days through providing a non-scheduled update [10]. In addition to compatibility problems, the non-disclosure of a renewed key cannot be guaranteed since the current touchlink preconfigured link key was also protected by an NDA but leaked anyway.

EZ-Mode Commissioning. EZ-Mode commissioning is an alternative commissioning procedure supported by all ZigBee 3.0 products. This commissioning procedure neither relies on inter-PAN frames nor a proximity check. Therefore, the attack procedures described in Section 4 cannot be adapted. EZ-Mode commissioning offers three options of securing the network key transport. In centralized security networks, the network key transport can be either protected by the publicly known default global Trust Center link key, or a link key derived from an install code. In distributed security networks, the network key transport is encrypted using the NDA-protected distributed security global link key.

[7]Please visit our website for the current state of software patches for the evaluated products: https://www1.cs.fau.de/content/zigbee-security-research

All three EZ-Mode commissioning options have serious drawbacks in terms of security and usability. Using the default global Trust Center link key makes the networks susceptible to attack scenarios where users are forced (e.g., by jamming) to recommission a node to the network in presence of the attacker. In this case the attacker would be able to recover the network key through eavesdropping on the commissioning procedure. Using the NDA-protected distributed security global link key is only secure as long as the key is not leaked. Thus, in the long run, the install code option is the only EZ-Mode commissioning option that is secure against the local attacker model described in Section 3.2. We note, however, that this option requires an extra effort from the users in terms of scanning (or entering) the install code using a mobile device app. This might constitute a serious usability problem, especially if already installed devices have to be recommissioned. For example, this can happen if a broken coordinator node has to be replaced. If the devices are difficult to reach, e.g., bulbs installed on the ceiling, the recommissioning might become quite cumbersome.

Limitations of our work. The ZigBee 3.0 specifications [20, p. 64] warn about supporting the touchlink command to join an end device to an existing centralized security network, which means that the hijack attack and network key extraction possibly cannot be performed on *centralized security networks*, if manufacturers heed this warning. However, these attacks work for distributed security networks. All other attacks work for both security network models.

7 RELATED WORK

The security of the ZigBee standard as well as the underlying IEEE 802.15.4 standard attracted much less attention in the academic research community compared to other wireless standards, such as Wi-Fi, Bluetooth or mobile telephony.

Sastry and Wagner [14] analyzed the security mechanisms of the IEEE 802.15.4 protocol. However, these mechanisms are not used in ZigBee. Wright [16] published the penetration testing tool KillerBee, which allows to sniff and analyze traffic of ZigBee and other IEEE 802.15.4-based networks. Wright also exposed that the network key of the then-current ZigBee standard was sent in clear text over the air. He demonstrated successful replay attacks using previously captured ZigBee traffic. The ZigBee Pro specification, released in 2012, addressed these security weaknesses. Goodspeed et al. [5] developed exploration tools to analyze the wireless attack surface of IEEE 802.15.4 networks. Armknecht et al. [1] present a formal security model for the ZigBee touchlink commissioning. Further papers [9, 15] cover security issues of ZigBee networks but these papers refer to security weaknesses concerning outdated ZigBee specifications and have not been evaluated with ZigBee-certified products.

Since the emergence of connected lighting systems in 2012, these systems have been subject to a number of security investigations. Dhanjani [4] published implementation weaknesses of the command authentication in the Philips Hue lighting system. He discovered that the secret whitelist token, which is required to authenticate the commands sent from the app (or website) to the bridge, is a hash of the MAC address of the controlling device. Chapman [3] obtained the firmware of LIFX light bulbs via a JTAG debugger and extracted cryptographic key material through reverse engineering of the firmware. Heiland [6] exposed vulnerabilities in the Osram Lightify system. Through reverse-engineering, he discovered that Wi-Fi credentials are stored in plaintext in the iOS Lightify Home app. Zillner et al. [26, 27] exposed security weaknesses in the ZigBee Pro specification. They showed that ZigBee-based lighting systems use publicly known fallback keys in the classical commissioning procedure for the initial key exchange, which allows the extraction of the network key. Ronen and Shamir [13] used the Philips Lux lighting system, which is the white-color variant of Hue, to build a covert channel for the exfiltration of data from an isolated environment. Also, Ronen et al. [12] exploited an implementation bug in Philips Hue bulbs that allowed to reset and to control these bulbs from a distance of a few hundred meters. In addition, they extracted cryptographic material, which secured the update process of the bulb's firmware, from the hardware using correlation power analysis. As a result, Ronen et al. were able to install a manipulated firmware image on Hue bulbs, and discussed the threat of a self-spreading IoT worm.

In contrast to related work on connected lighting systems, we do not analyze the products of a certain manufacturer but investigate the underlying standard. Also, we are the first to investigate the security mechanisms of the latest ZigBee standard, ZigBee 3.0.

8 CONCLUSION

Millions of IoT devices, including security-critical products that should be secured against local attackers, such as door locks and intruder alarm systems, use ZigBee for wireless low-power communication. In this work, we investigated the touchlink commissioning procedure, which is a commissioning option in the latest ZigBee specifications, ZigBee 3.0. We performed a security analysis of the touchlink commissioning procedure, in which we described active and passive attacks and evaluated their impact using our penetration testing framework Z3sec.

Our results conclude that attackers can thwart the availability of touchlink-enabled devices and can gain control over all nodes in the network. Already a single touchlink-enabled ZigBee device is able to expose the network key to an attacker, and therefore is sufficient to compromise the security of all nodes in the network, no matter how these nodes were added to the network. Thus, we warn about the adoption of touchlink commissioning in all future ZigBee 3.0 devices. To prevent these attacks, we recommend manufacturers of ZigBee-certified products to use EZ-Mode commissioning in combination with install codes.

Future work is required to analyze the security of the EZ-Mode commissioning in more depth. In addition, the design of a robust and suitable authentication infrastructure for IoT networks that interconnects function-constraint low-power nodes of multiple manufacturers without initial trust but with a high usability for non-expert users, is still a future challenge.

ACKNOWLEDGMENTS

The work is supported by the German Research Foundation (DFG) under Grant AR 671/3-1: WSNSec – Developing and Applying a Comprehensive Security Framework for Sensor Networks.

REFERENCES

[1] Frederik Armknecht, Zinaida Benenson, Philipp Morgner, and Christian Müller. 2016. On the security of the ZigBee Light Link touchlink commissioning procedure. In *International Workshop on Security, Privacy and Reliability of Smart Buildings*.

[2] Bastian Bloessl, Christoph Leitner, Falko Dressler, and Christoph Sommer. 2013. A GNU Radio-based IEEE 802.15.4 Testbed. *12. GI/ITG Fachgespräch Sensornetze* (2013), 37.

[3] Alex Chapman. 2014. Hacking into Internet Connected Light Bulbs. (July 2014). http://www.contextis.com/resources/blog/hacking-internet-connected-light-bulbs/

[4] Nitesh Dhanjani. 2013. Hacking Lightbulbs: Security Evaluation of the Philips Hue Personal Wireless Lighting System. (August 2013). http://www.dhanjani.com/blog/2013/08/hacking-lightbulbs.html

[5] Travis Goodspeed, Sergey Bratus, Ricky Melgares, Ryan Speers, and Sean W. Smith. 2012. Api-do: Tools for Exploring the Wireless Attack Surface in Smart Meters. In *45th Hawaii International Conference on Systems Science (HICSS-45 2012), Proceedings, 4-7 January 2012, Grand Wailea, Maui, HI, USA*. IEEE Computer Society, 2133–2140. DOI: http://dx.doi.org/10.1109/HICSS.2012.115

[6] Deral Heiland. 2016. R7-2016-10: Multiple Osram Sylvania Osram Lightify Vulnerabilities (CVE-2016-5051 through 5059). (July 2016). https://community.rapid7.com/community/infosec/blog/2016/07/26/r7-2016-10-multiple-osram-sylvania-osram-lightify-/vulnerabilities-cve-2016-5051-through-5059

[7] IEEE Computer Society. 2003. IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks Specific Requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). *IEEE Std 802.15.4-2003* (2003), 1–670. DOI: http://dx.doi.org/10.1109/IEEESTD.2003.94389

[8] Franciscus Wilhelmus Adrianus Alphonsus Van Leeuwen. 2014. Network discovery with touchlink option. (Feb. 27 2014). https://www.google.com/patents/WO2014030103A2 WO Patent App. PCT/IB2013/056,663.

[9] Olayemi Olawumi, Keijo Haataja, Mikko Asikainen, Niko Vidgren, and Pekka Toivanen. 2014. Three practical attacks against ZigBee security: Attack scenario definitions, practical experiments, countermeasures, and lessons learned. In *14th International Conference on Hybrid Intelligent Systems, HIS 2014, Kuwait, December 14-16, 2014*. IEEE, 199–206. DOI: http://dx.doi.org/10.1109/HIS.2014.7086198

[10] Philips. 2015. Friends of Hue - Update. (December 2015). http://www.developers.meethue.com/documentation/friends-hue-update

[11] J Picod, Arnaud Lebrun, and J Demay. 2014. Bringing software defined radio to the penetration testing community. In *Black Hat USA Conference*.

[12] Eyal Ronen, Colin O'Flynn, Adi Shamir, and Achi-Or Weingarten. 2017. IoT Goes Nuclear: Creating a ZigBee Chain Reaction. In *IEEE Symposium on Security and Privacy, S&P 2017*.

[13] Eyal Ronen and Adi Shamir. 2016. Extended Functionality Attacks on IoT Devices: The Case of Smart Lights. In *IEEE European Symposium on Security and Privacy, EuroS&P 2016, Saarbrücken, Germany, March 21-24, 2016*. IEEE, 3–12. DOI: http://dx.doi.org/10.1109/EuroSP.2016.13

[14] Naveen Sastry and David Wagner. 2004. Security considerations for IEEE 802.15.4 networks. In *Proceedings of the 2004 ACM Workshop on Wireless Security, Philadelphia, PA, USA, October 1, 2004*, Markus Jakobsson and Adrian Perrig (Eds.). ACM, 32–42. DOI: http://dx.doi.org/10.1145/1023646.1023654

[15] Niko Vidgren, Keijo Haataja, Jose Luis Patino-Andres, Juan Jose Ramirez-Sanchis, and Pekka Toivanen. 2013. Security Threats in ZigBee-Enabled Systems: Vulnerability Evaluation, Practical Experiments, Countermeasures, and Lessons Learned. In *46th Hawaii International Conference on System Sciences, HICSS 2013, Wailea, HI, USA, January 7-10, 2013*. IEEE, 5132–5138. DOI: http://dx.doi.org/10.1109/HICSS.2013.475

[16] Joshua Wright. 2009. KillerBee: Practical ZigBee Exploitation Framework. (2009). http://www.willhackforsushi.com/presentations/toorcon11-wright.pdf ToorCon 11.

[17] ZigBee Alliance. 2012. *ZigBee Light Link Standard Version 1.0 – Document 11-0037-10*.

[18] ZigBee Alliance. 2013. *Smart Energy Profile 2 Application Protocol Standard – Document 13-0200-00*.

[19] ZigBee Alliance. 2013. *ZigBee Home Automation Public Application Profile Version 1.2 – Document 05-3520-29*.

[20] ZigBee Alliance. 2016. *Base Device Behavior Specification Version 1.0 – Document 13-0402-13*.

[21] ZigBee Alliance. 2016. zigbee alliance Accelerates IoT Unification with 20 zigbee 3.0 Platform Certifications. (December 2016). http://www.zigbee.org/zigbee-alliance-accelerates-iot-unification-with-20-zigbee-3-0-platform-certifications/

[22] ZigBee Alliance. 2016. *ZigBee Cluster Library Specification Revision 6 – Document 07-5123-06*.

[23] ZigBee Alliance. 2017. The zigbee alliance to Unveil Universal Language for the IoT from CES 2017 — Making it Possible for Smart Objects to Work Together on Any Network. (January 2017). http://www.zigbee.org/the-zigbee-alliance-to-unveil-universal-language-for-the-iot-from-ces-2017-making-it-possible-for-smart-objects-to-work-together-on-any-network/

[24] ZigBee Alliance. 2017. ZigBee Certified Products. (2017). http://www.zigbee.org/zigbee-products-2/

[25] ZigBee Standards Organization. 2012. *ZigBee Specification – Document 053474r20*.

[26] Tobias Zillner. 2015. *White paper: ZigBee Exploited – The good, the bad and the ugly*. Technical Report. Cognosec.

[27] Tobias Zillner and Sebastian Strobl. 2015. ZigBee Exploited – The good, the bad and the ugly. (2015). https://www.blackhat.com/us-15/briefings.html#zigbee-exploited-the-good-the-bad-and-the-ugly Black Hat USA.