Securing Smart Home: Technologies, Security Challenges, and Security Requirements

Changmin Lee*, Luca Zappaterra*, Kwanghee Choi*†, and Hyeong-Ah Choi*
*Department of Computer Science, George Washington University, Washington, DC 20052. USA
†Korea Internet & Security Agency (KISA), Seoul, Korea
Email: *{clee1, lucaz, hchoi}@gwu.edu, †kchoisec@gmail.com

***Abstract*—Smart homes are gaining vast popularity as the most promising application of the emerging Internet of Things (IoT) technology. Exploiting the high level of connectivity present in current electronic devices (such as smartphones, tablets, and multimedia systems), smart homes provide innovative, automated and interactive services for residential customers through distributed and collaborative operations. As these types of networks become enormously popular, it is fundamental to provide the adequate level of protection against cyber-attacks for the residential customers. However, the resource-constrained nature of many of the devices present in a smart home environment, does not permit to implement the standard security solutions and therefore smart homes currently present security vulnerabilities. In this paper the security challenges and threats to the existing solutions suited for smart homes are examined in detail with the objective of fostering the development of practical solutions to secure the smart homes.**

I. Introduction

The Internet of Things (IoT) concept envisions the interaction and cooperation among smart objects surrounding us (such as home appliances, mobile devices, portable medical devices) to reach common goals [1], [2]. The IoT comprises of interconnected smart objects with access to the Internet, provided by networking technologies such as Bluetooth, ZigBee, Wi-Fi, aided by network gateways. The pervasive presence of these devices and their connectivity requirements generate very large amounts of data transmissions, which require guarantees of confidentiality, integrity, authenticity and reliability.

As one of the fastest growing fields of the IoT technology, Smart homes comprises of a network of smart devices which belong to different applications such as home automation, lighting control, climate control, entertainment and safety systems. Due to the critical private information a home environment contains, smart homes demand very stringent cyber-security requirements. Existing early-stage smart home solutions have shown significant vulnerabilities [3], [4], demonstrating the immaturity of current security solutions to be released for large market penetration. Consequently, the security aspects of smart homes need to be tackled and solved before this technology becomes ubiquitous.

In this paper, we conduct an analysis of the main challenges and security threats present in smart home networks. The results of the analysis are then used to draw the fundamental requirements needed for providing secure and confidential operations in smart homes.

The remainder of the paper is organized as follows. In Section II, the technologies composing a smart home are discussed in detail, describing applications, devices, operating systems, and communication protocols. Following Section III discusses the main challenges present in securing smart homes. In Section IV, the security threats present in the communication protocols are illustrated with case studies, defense strategies and countermeasures. Section V presents the security requirements of smart homes. In Section VI, concludes the paper and provides future research directions.

II. Smart Home Technology

A smart home comprises of a multitude of connected devices belonging to different application areas. These devices are characterized by heterogeneous hardware and software resources, and they support different communication technologies. By coexisting, interacting, and cooperating among each others, these devices form a distributed heterogeneous network. As an example, Figure 1 shows a typical smart

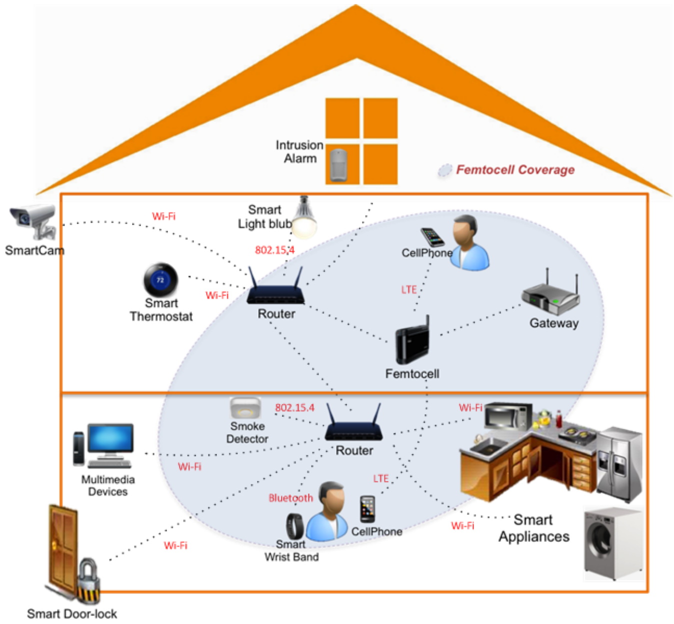

Fig. 1: An example of smart home consisting of connected devices belonging to different applications and gateways providing connectivity with the Internet.

978-1-4799-5890-0/14/$31.00 ©2014 IEEE

home scenario, consisting of many connected devices belonging to different applications, communicating among each others using different technologies and connected to few gateways/routers which provide connectivity to outside networks such as Internet.

A. Applications

The smart home concept encloses multiple applications belonging to the different areas, interacting with each others.

Lighting control: Intelligent home lighting systems provide automated lighting control through LED lights and controllers detecting ambient conditions such as the presence of users or sunlight. The system automates the actions of turning on and off the lights and controlling their brightness according to the user's preferences and activities or energy savings rules.

Appliance control: Home appliances such as refrigerators, ovens, and washing machines contain embedded devices for the control of their functioning. In the past, these embedded devices constituted stand-alone systems, each providing dedicated control only for a single appliance. The significant dropping in prices of the transceiver chips, with the consequent inclusion of communication functionalities into these embedded devices, have created unlimited possibilities for cooperative and automated appliance control. In example, high-energy consuming appliances such as washing machines and ovens could schedule operations during off-peak energy rates. Also, washing and drying machines can share laundry setting information.

Entertainment: A typical house has one or more entertainment center(s) consisting of different devices (e.g., Digital TV, DVD player, satellite decoder, digital multimedia receiver) together with multiple media players such as tablet PCs, smartphones, MP3 players. Entertainment systems in smart homes provide connectivity, access to shared resources and distribution of content according to users' preferences.

Safety system: A safety system of a smart home consists of smoke detectors, zone intrusion detectors, burglar alarms, surveillance cameras, interconnected and integrated with the communication infrastructure and the information systems. Alarms are properly and timely reported to the intended receiver(s) which could be the residents or a police station.

Climate control: Heating, ventilation and air conditioning (HVAC) can be integrated and controlled jointly in an automated way, with the ultimate goal of providing customized climate control while saving energy. Room temperatures can be regulated based on human presence, in example providing at nighttime higher temperatures in the bedrooms, or lowering the climate control when the house is empty and activating it just before the residents come back home.

Assisted living: In a smart home assisted leaving and telecare can be provided for elder people to assist and monitor them, with the ultimate goal of permitting them to live longer in their houses. Sensors can recognize the activity of a person and assistance can be provided accordingly. In example, detecting a person waking up should automatically turn on the lights; also unexpected behaviors such as a person falling on the floor, shall be detected and reported to the emergency units promptly.

B. Devices

Several types of devices can coexist in a smart home. Possibilities for new smart devices are endless, since any residential device with the addition of intelligent computational capabilities can become part of a smart home. Here the main devices currently available are listed, grouped by target application.

Lighting control: Light bulbs, light strips.

Appliance control: laundry machines, refrigerators, ovens.

Safety system: smoke detectors, intrusion detection devices, security cameras, smart door-locks.

Entertainment: Smart-TVs, set top-boxes, media players, laptops, wireless speakers.

Health and assisted living: smart wrist bands, portable ECGs, pulse oximeters.

Network devices: gateways, routers, network storage devices, mobile phones, printers.

C. Operating Systems

Due to the size, energy, computation and storage limitations typical of the embedded systems implemented in the majority of the devices in a smart home, their operating systems (OSs) must be extremely lightweight, while supporting the rich set of application, communication and security features needed. There exist a few operating systems for IoT devices that can currently be adopted for research and development purposes. Many researchers and developers are working for applying their innovative solutions into to these OSs, with the ultimate goal of testing them for production. Following, the main OSs applicable for smart home devices are presented.

Contiki *[5]*: this open source OS for IoT applications which provides connectivity and applications' support for low-cost, low-power micro-controllers. Contiki represents the most adopted solutions for developing IoT solutions, It is written in C language and it has been ported to a number of microcontroller architectures, including the Texas Instruments MSP430, Atmel AVR, and the ESB platform [5]. Probably influenced by its popularity, there have been numerous studies of the protocol implementation vulnerabilities of Contiki [6].

Tiny OS *[7]*:this free and open-source tool consists of a component-based OS and a development platform targeting wireless sensor network (WSN) applications. TinyOS is implemented using nesC programming language as a set of cooperating tasks and processes. Started as a collaboration between the University of California, Berkeley in co-operation with Intel Research and Crossbow Technology, it has since grown to be an international consortium, the TinyOS Alliance.

RIOT OS *[8]*: it is a microkernel-based OS which match the specific software requirements of typical IoT devices. Thanks to its modular implementation, RIOT OS guarantees minimum memory usage and permits ad-hoc customizations and configurations to meet the specific application requirements. Because of the minimized kernel size, RIOT OS requires only

TABLE I: Specifications of smart home devices

Device Type	Chipset	Core Freq.	RAM	Flash Memory	Power	Networks Protocols
iPhone	A7x Quad-core Processor	1.7Ghz	2GB	Up to 128GB	Battery	Wi-Fi, Bluetooth, NFC
Nest Learning Thermostat	ARM Cortex-A8	800Mhz	512MB RAM	2GB	Battery	Wi-Fi (802.11)
Nest Smoke Detector	ARM Cortex-M4	100Mhz	128KB RAM	512KB	Battery	Wi-Fi (802.11)
	ARM Cortex-M0	48Mhz	16KB RAM	128KB		
NETGEAR Router	Broadcom BCM4709A	1.0Ghz	256MB	128MB	AC Power	Wi-Fi (802.11)
Samsung Smart TV	ARM-based Exonys SoC	1.3Ghz	1GB	N/A	AC Power	Wi-Fi (802.11)
Samsung SmartCam	GM812x SoC	Up to 540Mhz	N/A	Up to 64GB	AC Power	Wi-Fi (802.11)
Elster REX2 Smart Meter	Teridian 71M6531F SoC	10Mhz	4KB	256KB	Battery	ZigBee (802.15.4)
Philips Hue Light bulb	TI CC2530 SoC	32Mhz	8KB	Up to 256KB	Battery	ZigBee (802.15.4)
Fitbit Smart Wrist Band	ARM Cortex-M3	32Mhz	16KB	128KB	Battery	Bluetooth LE
Sensor Devices	Microcontroller	4 − 32Mhz	4 − 16KB	16 − 128KB	Battery	ZigBee, Wi-Fi, Bluetooth

few hundreds bytes of RAM and storage. Although RIOT OS is still in development, it has the potential to become the preferred OS for IoT devices, due to its lightness and customization ability.

In addition to the solutions listed above, the IoT devices currently on the market implement proprietary OSs. Although developed mainly for research purposes and supported by voluntary contributors, the open source OSs listed above represent mature and valid software solutions possibly ready to be customized to match the specific security requirements of smart devices to be developed.

D. Communication Protocols

The heterogeneity of the hardware and software components in a smart home also reflects in the communication protocols available. Different solutions are used to transport information between devices, depending on traffic characteristics, device capabilities, and surrounding environment. The main communication protocols used in a smart home are briefly described next, grouped according to the OSI model classification.

1) Physical and Data Link Layers:

IEEE 802.15.1 Standard: it defines the wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPANs) targeting high-speed data transfers and multimedia distribution for home entertainment over short distances around 10 meters. It operates using frequency hopping spread spectrum (FHSS) to combat interference and jamming, achieving a maximum data rate of 1Mb/s over 1MHz channels in the unlicensed industrial, scientific and medical (ISM) band at 2.4 GHz. The Bluetooth and Bluetooth LE (Low-Energy) protocol architectures build on top of the IEEE 802.15.1 PHY and MAC layers. The IEEE 802.15.1 Standard addresses the network security aspects of authentication (through a challenge-response 128-bit private key scheme) and encryption (through variable size up to 128 bits private key), without addressing message integrity issues.

IEEE 802.15.4 Standard: it specifies MAC and PHY protocols for low-rate WPANs, targeting at networks characterized by devices such as sensors and embedded devices with limited traffic, low energy available and constrained memory and processing capabilities. The standard achieves a maximum data rate of 250 Kb/s transmitting on 5MHz channels in the 2.4 GHz ISM band using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). The ZigBee high level communication protocol suite is based on the underneath IEEE 802.15.4 physical and data link layers. Eight different suites provide security guarantees for the applications, ranging from adopting encryption only (AES-CTR), authentication only (AES-CBC-MAC), or encryption and authentication (AES-CCM).

IEEE 802.11 Standard: this technology provides MAC and PHY specifications for high-rate communications in Wireless Local Area Networks (WLANs) with ranges from 20 to 250 meters. In its 802.11a and 802.11g specifications it achieves 54 Mb/s data transmissions using CSMA/CA on 20 MHz channels, through the use of Orthogonal Frequency Division Multiplex (OFDM) modulation. The latest developments of 802.11n and 802.11ac significantly increase the data rates, setting the theoretical upper bounds to 150 and 866.7 Mb/s respectively. These improvements are reached through the use of larger bandwidths through channel aggregation up to 40 MHz for 802.11n and and 160 MHz for 802.11ac in the 2.4, 3.6, 5 GHz frequency bands and by adding multiple-input multiple-output (MIMO) antenna functionalities. In its first inception, IEEE 802.11 Standard used the Wired Equivalent Privacy (WEP) method to encrypt data, which was shown to be severely weak. In later implementations, robust solutions such as Wired Equivalent Privacy (WPA) and 801.11i emerged.

2) Network and Transport Layers:

Network layer protocols for smart home applications can either belong to custom designed solutions to address the specific application requirements (such as Zigbee or Bluetooth network layer protocols) or implement IP-based networking functionalities using the underneath physical and data layer solutions described above. In this context, the latter case has gained increasing interest due to the advantage of supporting IPv6 functionalities such as large address space, stateless and stateful address configuration, needed by the IoT applications. Protocols such as IPv6 over Low power Wireless Personal Area Networks (6loWPAN) [9], Routing Protocol for Low power and Lossy Networks (RPL) [10], and Multicast Protocol for Low power and Lossy Networks (MPL) [11] are examples of future adaptation and networking protocols taylored for IoT applications, including smart homes.

Regarding the transport layer, UDP is preferred for resource-constrained devices, since it saves power by going to sleep after transmitting a packet, oppositely to TCP, which enforces to stay awake to process acknowledgments. At the same time, UDP lacks in reliability, which may be required at different levels, depending on the specific application considered. The reliability requirements of the transport protocol depend on the target application. For example, different mechanisms could be implemented for packet-based applications (which require all packets be received reliably at the destination) versus event-based applications (which require events, and not necessarily all individual packets, be reliably reported to the destination). An event-based application might call for less-complex transport mechanisms.

3) Application Layer:

The main application protocols for IoT and smart home environments are summarized below.

eXtensible Messaging and Presence Protocol (XMPP) *[12]*: this open-standard protocol implements a message-oriented middleware based on XML and it is widely tested and adopted for IoT applications such as smart grids and remote monitoring.

Constrained Application Protocol (CoAP) *[13]*: this software protocol is targeted for resource-constrained electronics devices that need to be controlled or supervised remotely, through Internet-based networks. CoAP is designed to minimize the complexity of mapping with HTTP, while also meeting specialized requirements such as multicast support, low overhead, and implementation simplicity.

MQ Telemetry Transport (MQTT): this connectivity protocol represents an extremely lightweight publish/subscribe messaging transport designed for low complexity, low power and low footprint implementations. It runs on connection-oriented transport layer protocols such as TCP or on non-TCP/IP networks through its MQTT-S variant.

III. Security Challenges

Smart homes require very stringent security requirements, due to the importance of the private information a home environment contains. The main challenges present in a smart home that prevent the use of standard security mechanisms adopted in conventional networks are described below.

Resource Constraints: as shown in Table I, majority of smart home devices are designed to work with low-power and reduced-size hardware, which constrain their computing performance and storage capabilities. In [14], authors tested current security mechanisms on a sensor device with 8MHz CPU, 10KB of RAM, and 48KB program memory, proved that current security mechanisms are not feasible in small sensor devices. Public key algorithm such as RSA and ECC are very intensive and requires many multiplication instructions to perform a single security process.

Heterogeneous Communication Protocols: the different protocols possibly used to inter-connect the devices in a smart home require the use of intermediate gateways, which pose a major limitation for the implementation of end-to-end security solutions between end-devices in the smart home and the Internet applications.

Unreliable communications: majority of the communication protocols described before do not guarantee reliability of packet delivery. In fact, packets could fail or being damaged due to collisions or highly congested nodes. Retransmissions and error handling algorithms require large overhead, not tolerable in low-power networks devices [15].

Energy Constraints: many devices in a smart home are operated with battery power and therefore with limited energy availability for their communications, storage and computations. First, energy limitations provide vulnerability to resource depletion attacks that force devices to stay awake,consuming. Secondly, the implementation of security methods add computations, storage requirements and communication overheads, all drastically consuming the available energy.

Physical Access: in a smart home devices can be left unattended all the time, becoming easy targets of tampering attacks. If an attacker obtains physical access of a device in a smart home, he may be able to extract from a device the pre-defined encryption keys and other sensitive information.

IV. Analysis of existing security threats

Existing security threats of wireless networks as well as new security threats apply to smart homes. The main threats to each OSI communication layer are presented in this section and summarized in Table II.

A. Attacks to the Physical Layer

Jamming and tampering represent the two main security threats to the PHY.

Jamming: it consists of an emission of radio signals with the goal of disturbing and/or disrupting the communication of a victim device. While wireless interference is unintentional, jamming is intentional and focus on a specific target [16]. In the worst case possible, an attacker with a powerful jamming source can disrupt the entire communications of a victim network. Also, an attacker could make quickly drain the battery of target devices by intentionally disrupting their data transmission and make them repeatedly retransmit. According to [17], one bit transmitted in a WSN consumes about as much power as executing 800-1000 instructions. Thus, jamming could lead to serious denial of service (DoS) of smart home devices.

Tampering: giving an attacker physical access to devices, opens up a numerous attacks listed below:

- *Malicious Code Injection*: it consists of the injection of malicious software through the debugging interface of the device. Then injected malicious code could disrupt the entire smart home network since the device with malicious code is already deployed in the network. Alternatively, an attacker may obtain all information in the home network by injecting

TABLE II: Security threats from each protocol layer

Layer	Protocols	Threats & Attack Framework
Application	CoAP, XMPP, MQTT	XMPPloit(Framework)
Transport	TCP, UDP	UDP Flooding, TCP SYN Flooding, De-synchronization
Network	MPL, RPL, 6LoWPAN	KillerBee(Framework), Black-hole Attack, Change Rounting Information, Packet Capture & Injection, Selective-Forwarding, Sinkhole, Hello Flood, Wormhole, Sybil, Tiny Fragmentation
Data Link	802.15.4, 802.11, 802.15.1	KillerBee(Framework), GTS Attack, Back-off manipulation, ACK attack
Physical	802.15.4, 802.11, 802.15.1	Jamming, Tampering

malicious software that has code for seamlessly transmitting all information outside of the protected network.

- *Extraction of Security Information*: sensitive information such as pre-installed encryption keys can be extracted from devices by stealing the actual driver or connecting to a device. An example of a successful attack extracted the firmware of a smart meter using a debugging cable [18].

- *Duplication of a Device*: a malicious manufacturer could duplicate the features of a genuine device including hardware, software and configurations. A malicious device installed in a smart home could run malicious software to manipulate a target genuine device, or degrade the functionalities of other devices. In [19] authors presented their successful hack into an iPhone, exploiting the use of a malicious duplicate charger which installed a trojan into the device software.

B. Attacks to the Data Link Layer

Often the data link layers represent the most vulnerable algorithms in the communication stack. The main attacks to the data link layer are described below.

KillerBee: this framework provides a set of tools for exploiting vulnerabilities in ZigBee and IEEE 802.15.4 networks. Killerbee simplifies sniffing, injecting traffic, packet decoding and manipulation, as well as recon and exploitation. Numerous attacks can be carried out using KillerBee such as PANId conflict, replay attack, packet capturing, and network key sniffing [20].

GTS Attack: Guaranteed Time Slot (GTS) attack is is based on the properties of the IEEE 802.15.4 superframe organization in beacon-enabled operational mode. Authors in [21] explain that the GTS slots create a vulnerable point which can allow an attack to disrupt the communication between a device and its coordinator (gateway). An adversary can obtain the allocated GTS times, and be able to create interference at any of these moments. The interference will cause collision and corruption of the data packets between devices, and make target nodes to retransmit data packet repeatedly.

Back-off manipulation: this attack is possible in CSMA/CA based networks such as IEEE 802.11 and IEEE 802.15.4 networks. A malicious device constantly choose a small back-off interval for contention using the Distributed Coordination Function (DCF) [22], not giving chances for medium access to the victim devices.

ACK attack: an adversary can apply an ACK attack to a target smart home environment by eavesdropping the wireless channel. An adversary may block the receiver device from taking the transmitted packet, then, it can mislead the sender device by sending a fake ACK that it comes from the receiver device [23].

C. Attacks to Network Layer

Due to the heterogeneous communication protocols, smart home network is inherently vulnerable to the different types of attacks such as flooding, spoofing, network sniffing, data capturing and modification, DoS. Many network layer attacks not specific to the smart home environment have been documented in previous sudies [16], [24]–[29]. Here a specific attack to the RPL networking protocol, used for IoT networks is described.

Black hole attack on RPL: this attack targets the RPL implementation of ContikiOS. In [6], the authors demonstrated a black hole attack which is initiated by a compromised node that act maliciously in the network dropping the packets that are routed through it, causing disruptions in the data flow of the network. A black hole attack can be effectively disguised and may lead to an attacked network behave very similar to a health network. It is important to notice that the attack has been successfully carried out only on in ContikiOS based devices, and other OS such as Tiny OS, RIOT OS, having alternative implementations of the RTL protocol do not show vulnerability towards this attack.

D. Attacks to the Transport Layer

No specific attacks to the transport layer of a smart home network are documented, although general well-known attacks to the transport layer protocols such as flooding and de-synchronization [29] can be applied targeting a smart home victim network.

E. Attacks in Application Layer

Many of the a application layer attacks known in literature may apply also to smart homes. Here a specific attack that can be conducted targeting smart homes security is reported.

***XMPPloit** [30]*: this is an command-line exploit tool to attack XMPP connections which exploits vulnerabilities at the client and server side utilizing the XMPP protocol. XMPPloit can force a smart home client device not to encrypt its communications, so that an attacker may read and modify them while they are transmitting.

V. Security Requirements for Smart home

Following the observations from the previous sections, the main security requirements of a smart home are derived and described below.

User Authentication: hundreds of devices with Internet connectivity, requiring software updates, security patches, and data exchanges will be deployed in smart homes. All these process need to be performed by authorized users only: without strong user authentication, a smart home won't be safe from attackers.

Device Authentication: As described in Section IV, a smart home network must be protected from compromised nodes' attacks. Device authentication shall provide the ability to identify legitimate devices from unauthorized devices in the smart home network.

Network Monitoring: since DoS attacks can target the network layer protocols running on a smart home network, it is fundamental to have an intrusion detection system and monitoring tool to detect network intrusions and report traffic anomalies.

Secure Key Management: since some of the sensor devices are deployed with pre-installed network keys, it is required to have a secure key-management scheme to protect the smart home from attackers that compromised devices inside the network.

Physical Protection: It is a known fact that electronic devices and are often left unattended, becoming vulnerable to tampering attacks. Thus, physical protection is one of the important requirements for smart homes. Tamper resistant devices or anti-reverse engineering schemes would be solutions against tampering attacks.

VI. Conclusion

In this paper the major technologies, security challenges, and threats in smart home networks is presented. Analyzing the applications, devices, and technologies available for the implementation of smart homes networks, the main security challenges are identified in implementing performant security algorithms according to the resource and energy limitations of the devices, guaranteeing end-to-end security in heterogeneous and unreliable wireless networks and finally physically securing the smart home against tampering of devices. Following, the existing threats present in the current networking technologies adopted for devices' communications in smart homes are described. Finally, from the study conducted, the main security requirements for smart homes are derived. These identified requirements constitute the fundamental basis for researching and developing new security algorithms tailored for the specific security characteristics of smart homes.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[3] "Belkin patches massive smart home device vulnerability." http://news.yahoo.com/belkin-patches-massive-smart-home-211118220.html.

[4] "Hacking home automation systems through your power lines." http://www.wired.com/2011/08/hacking-home-automation.

[5] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pp. 455–462, IEEE, 2004.

[6] K. Chugh, A. Lasebae, and J. Loo, "Case study of a black hole attack on 6lowpan-rpl," in *SECURWARE 2012, The Sixth International Conference on Emerging Security Information, Systems and Technologies*, pp. 157–162, 2012.

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

[8] E. Baccelli, O. Hahm, M. Wählisch, M. Gunes, T. Schmidt, *et al.*, "Riot: One os to rule them all in the iot," 2012.

[9] J. Hui, D. Culler, and S. Chakrabarti, "6lowpan: Incorporating ieee 802.15. 4 into the ip architecture," *IPSO Alliance White Paper*, 2009.

[10] T. Winter, "Rpl: Ipv6 routing protocol for low-power and lossy networks," 2012.

[11] J. Hui and R. Kelsey, "Multicast protocol for low power and lossy networks (mpl)," 2013.

[12] P. Saint-Andre, "Extensible messaging and presence protocol (xmpp): Core," 2011.

[13] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, "Constrained application protocol (coap), draft-ietf-core-coap-13," *Orlando: The Internet Engineering Task Force–IETF, Dec*, 2012.

[14] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz, "Comparing elliptic curve cryptography and rsa on 8-bit cpus," in *Cryptographic Hardware and Embedded Systems-CHES 2004*, pp. 119–132, Springer, 2004.

[15] J. Sen, "A survey on wireless sensor network security.," *International Journal of Communication Networks & Information Security*, vol. 1, no. 2, 2009.

[16] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A survey on jamming attacks and countermeasures in wsns," *Communications Surveys & Tutorials, IEEE*, vol. 11, no. 4, pp. 42–56, 2009.

[17] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in *ACM SIGOPS operating systems review*, vol. 34, pp. 93–104, ACM, 2000.

[18] N. Lawson, "Reverse-engineering a smart meter." http://rdist.root.org/2010/02/15/reverse-engineering-a-smart-meter/.

[19] C. NGAK, "Hacked iphone chargers could let snoops spy on devices." http://www.cbsnews.com/news/hacked-iphone-chargers-could-let-snoops-spy-on-devices/.

[20] J. Wright, "Killerbee - practical zigbee exploitation framework," in *Toorcon11*, 2009.

[21] R. Sokullu, I. Korkmaz, and O. Dagdeviren, "Gts attack: An ieee 802.15. 4 mac layer attack in wireless sensor networks," *International Journal On Advances in Internet Technology*, vol. 2, no. 1, pp. 104–114, 2009.

[22] S. Radosavac, A. A. Cárdenas, J. S. Baras, and G. V. Moustakides, "Detecting ieee 802.11 mac layer misbehavior in ad hoc networks: Robust strategies against individual and colluding attackers," *Journal of Computer Security*, vol. 15, no. 1, pp. 103–128, 2007.

[23] Y. Xiao, S. Sethi, H.-H. Chen, and B. Sun, "Security services and enhancements in the ieee 802.15. 4 wireless sensor networks," in *Global Telecommunications Conference, 2005. GLOBECOM'05. IEEE*, 2005.

[24] A. Le, J. Loo, A. Lasebae, M. Aiash, and Y. Luo, "6lowpan: a study on qos security threats and countermeasures using intrusion detection system approach," *International Journal of Communication Systems*, vol. 25, no. 9, pp. 1189–1212, 2012.

[25] R. Riaz, K.-H. Kim, and H. Ahmed, "Security analysis survey and framework design for ip connected lowpans," in *Autonomous Decentralized Systems, 2009. ISADS '09. International Symposium on*, March 2009.

[26] L. Wallgren, S. Raza, and T. Voigt, "Routing attacks and countermeasures in the rpl-based internet of things," *International Journal of Distributed Sensor Networks*, vol. 2013, 2013.

[27] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks*, vol. 1, no. 2, pp. 293–315, 2003.

[28] T. Roosta, S. Shieh, and S. Sastry, "Taxonomy of security attacks in sensor networks and countermeasures," in *The First IEEE International Conference on System Integration and Reliability Improvements*, 2006.

[29] A. Wood and J. A. Stankovic, "Denial of service in sensor networks," *Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[30] "Xmpploit." http://tanguy.ortolo.eu/blog/article69/xmpploit-explained.