A System to Save the Internet from the Malicious Internet of Things at Home

Lukas Braun
Munich University of Applied Sciences
MuSe – Munich IT Security Research Group
Munich, Germany
email: lukas.braun@muse.bayern

Hans-Joachim Hof
Technical University of Ingolstadt
CARISSMA – Center of Automotive Research on Integrated Safety Systems and Measurement Area
Ingolstadt Research Group Applied IT Security Ingolstadt,Germany
email: hof@thi.de

Abstract— **Botnets are a big hassle for the Internet. A recent attack by the Mirai botnet showed how easy it is to exploit Internet of Things devices and use them for malicious activities, e.g., for sending spam or executing Distributed Denial of Service attacks. Hence, increasing protection of Internet of Things (IoT) devices as well as increasing protection against malicious Internet of Things devices is an important challenge. Many of the Internet of Things devices used in the Mirai botnet are located in smart homes (e.g., surveillance cameras). This paper presents a novel smart home security system that raises the bar for an attacker by separating different classes of Internet of Things devices in a smart home from each other, as well as separating other devices within the smart home network (e.g., desktop computers) from Internet of Things devices. Amongst other measures, the smart home security system enforces strict security policies on outgoing communication of Internet of Things devices. By doing so, the proposed smart home security system is able to limit the effect hacked Internet of Things devices in a smart home have on the Internet.**

Keywords- Secure Smart Home; Internet of Things security;

I. INTRODUCTION

In October 2016, a gigantic botnet, the Mirai botnet, was used for various attacks on the Internet. Amongst other things, the Mirai botnet attacked parts of core Internet services, resulting in outages or slow responses from popular websites like Twitter, Spotify, and Reddit [1]. A notable aspect of the Mirai network is that it maliciously uses a large number of IoT devices in smart homes, e.g., DVRs (Digital Video Recorders) and surveillance cameras. The high number of malicious IoT devices allows the Mirai botnet to achieve an attack load of 1.2 Tbps (Terabit per second). Such intensive traffic renders even advanced protection useless mechanisms or makes using them very expensive.

The IoT connects IoT devices with each other and with gateways, infrastructure, and backend services. IoT devices are things from the physical world that are equipped with sensors and/or actuators. As a whole, the IoT extends the cyberspace to the physical world by sensing and acting in the physical world via IoT devices. IoT devices are known for being vulnerable to attacks. A study conducted by HP in 2014 found serious security flaws in IoT devices, e.g., 70% of IoT devices did not encrypt communication to the Internet and local network and 60% of IoT devices raised security concerns with their user interface [2]. IoT devices may be used in different domains and for different applications, e.g., in manufacturing, commercial building automation and the like. This paper focuses on IoT devices used in smart homes by private users. IoT devices for private smart homes often have a low security level due to three reasons: Reason number one is a huge cost pressure on IoT device manufacturers by the market. In such a situation, security, as a non-functional requirement that results in no product feature, may be the number one requirement to be dropped to save money during development of IoT devices. The second reason is the user. Users of IoT devices in private smart homes are usually not well educated regarding IT security. Hence, a thorough security analysis and a rigorous hardening of IoT devices is not expected in this domain. Reason number three is the limited user interface of a typical private smart home IoT device. Security configuration by the user may not be intended because of the lack of a suitable user interface or management protocol. Taking into consideration the low security level of IoT devices in private smart homes, the powerful network connection of smart home, and the missing network based security controls, these IoT devices are valuable attack targets for botnet owners.

The rest of this paper is structured as follows. Section II presents the state of the art in smart home security as well as related work on this topic. Section III presents the reference architecture of the work presented in this paper. The section also states important security requirements the smart home security system presented in this paper must fulfill. Section IV gives an overview on the proposed smart home security system and presents selected aspects in more detail. Section V reports on the ongoing implementation of the prototype. Section VI concludes the paper.

II. STATE OF THE ART IN SMART HOME SECURITY AND RELATED WORK

In a typical private smart home network, mostly two different security methods are used: A firewall runs on the internet gateway (home router) to prevent attacks from the internet and some endpoints are secured using security controls like virus scanners and personal firewalls. However, endpoint security controls are typically only used on desktop computers. Other devices like smart TVs, surveillance

 ISBN: 978-1-61208-582-1

cameras, or DVRs usually do not have security controls in place, albeit nowadays these devices are often based on traditional operating systems like Windows or Linux. A typical private smart home network does not implement security controls to monitor or restrict internal network traffic, or to separate devices from each other. Hence, one vulnerable device in a private smart home network may be enough for an attacker to spread malware throughout the network or to hack into other systems. In contrast, many companies are using network-based security controls to separate network traffic, e.g., based on the criticality of the traffic. However, this approach needs an in-depth network engineering that is likely not happening in home networks because the average smart home network owner neither has the necessary experience with secure network nor the willingness to pay for network engineering services. This paper presents a smart home security system that implements advanced network security controls and is suitable for private users. Users do not need special security training to use the smart home security system.

Many existing solutions for smart homes are focused on special aspects or special applications of smart home security, e.g., they focus on smart homes as part of the smart grid [7-10] These solutions are not suitable to protect the Internet from IoT devices in smart homes. Other publications like [13] focus on special network protocols used in current building automation systems, e.g., ZigBee. This paper assumes that IoT devices do not use special communication protocols, but rather are integrated using WiFi. The Universal Home Gateway presented in [11] is a similar approach to smart home security as presented in this paper. However, the approach of [11] is based more on services to be implemented on the home router than on having a smart network filtering available. The smart home security system presented in this paper is compatible with legacy IoT devices, allowing them to also participate in the network. Also, devices being aware of the proposed smart home security system do not need to provide code for services running on a home router as in [11]. They only need to provide a special kind of attribute certificate. Hence, the approach presented in this paper is more flexible.

III. SMART HOME REFERENCE ARCHITECTURE AND SMART HOME SECURITY REQUIREMENTS

Figure 1 shows the smart home reference architecture used for the work presented in this paper. It is based on our previous work [3]. The smart home consists of several networks, e.g., a home automation network (e.g., based on Z-Wave, ZigBee, KNX, or any other proprietary home automation protocol), and a home network (based on WiFi or Ethernet). Gateways (GW) may interconnect these networks. The smart home security system presented in this paper is implemented in the home network (based on WiFi or Ethernet) as many recent IoT devices for smart homes support WiFi (at least via a gateway). A home router typically controls the home network. The home router also connects the home network to the Internet. The range of the home router may be extended by so called range extenders (not shown in Figure 1). The reference shows different classes of devices typically used in private smart homes (e.g., smartphones, tablets, home entertainment equipment, household appliances, etc.). These classes are essential for the design of the presented system and are presented in more detail in Section IV.A.

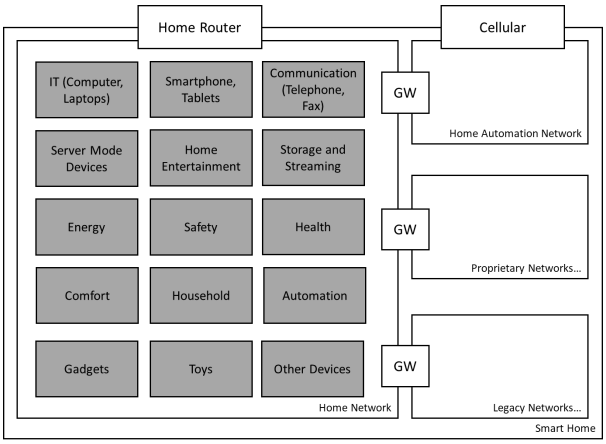

Figure 1. Smart Home Reference Architecture

The following security requirements are considered essential for security in a private smart home implementing the reference architecture:

- R1: IoT devices are only allowed to communicate with intended communication partners.
- R2: IoT devices are assigned to classes based on their application area and communication properties.
- R3: Communication between classes is only allowed based on well-defined security policies.

R1 ensures that IoT devices can only communicate with a known number of external partners. For example, a Sony smart TV may only be allowed to communicate with communication partners in the domain sony.com as well as streaming providers like Netflix. This drastically reduces the number of attackable systems if this IoT device gets hacked. R3 separates IoT devices of different device classes from each other. Together with R2, this enables the definition of generic rules for intra-home network communication. For example, a Playstation 3 in the home entertainment class may only get access to a media server in the storage and streaming class, but no access to devices in the smartphone class, whereas a smart phone from the smart phone class may be allowed to initiate communication with all other device classes for content streaming.

Non-security requirements include usability of the proposed system. Usability is very important in private smart homes as inexperienced users are considered the default users. The system follows the design guides presented in [4-6], especially design guidelines G1 (understandability, open for all users), G3 (no jumping through hoops), G4 (efficient use of user attention and memorization capabilities), G6 (security as default), and G7 (fearless system) are obeyed in design of the smart home security system. Compliance with these guidelines is achieved by automating as many tasks as possible, hence requiring as little user interaction as possible. If smart home IoT devices are aware of the proposed

 ISBN: 978-1-61208-582-1

smart home security system, the only user interaction is a confirmation request for the addition of a new device. The rest of the configuration process is hidden from the user. This allows even users that are unexperienced in IT security or even IT to use the proposed smart home security system.

IV. DESIGN OF THE SMART HOME SECURITY SYSTEM

The home router is the central point for enforcement of security policies for the smart home security system. It enforces network security policies on a per-class and per-device basis. Security policies allow or forbid certain communication partners. They state allowed traffic patterns. Communication partners may, e.g., be described as a class (only intra-home network communication), a domain, a subdomain, or an IP address range. Wildcards may be used (but should be avoided if possible). See Section IV.B for details on the hierarchical ordering used in the definition of communication partners. Using this approach, total transparency is achieved, as all communication partners of IoT devices must be registered at the home router, and the home router can list all communication partners for each device to the user. For example, if an IoT device uses a third party IoT platform and sends data to this platform, it is necessary to state this in the security policy for this device; otherwise, no connection with the IoT platform is possible. Hence, a user buying a device from a German IoT company may learn that this device regularly communicates with servers in mainland China by inspecting the security policies on the home router. Transparency enables the customer to only buy IoT devices that satisfy their privacy needs (e.g., IoT devices that do only communicate with communication partners in Europe, where the General Data Protection Regulation applies).

The attacker model for the proposed smart home security architecture considers IoT devices to be trusted at integration time. Automated detection of malicious IoT device manufacturers is out of scope of this paper. A malicious IoT device manufacturer usually has full control over the IoT device and encrypted communication with the manufacturer is not suspicious (software updates may be an expected feature). Hence, there is not much possibility to detect or avoid such an attack.

A. Classification of Smart Home Devices

During the integration into the network the device get a class assigned and relevant security policies are retrieved. Available classes are described in more detail in Table 1. They are based on currently existing IoT devices in typical smart home use cases.

TABLE I. DEVICE CLASSES

Class	Example / Description	Challenge / properties
IT	Classical IT devices like computers or laptops	Typical devices in this class are multipurpose, hence it is not possible to describe typical traffic patterns or have a full description of communication partners. As there are typically already many security controls installed (virus scanner, personal firewall, …), devices in this class are allowed to make generous use of wildcards when stating security policies. However, existing filter lists for websites and the like may be used.
Smartphones, Tablets	Smartphone, tablet	Similar to class "IT". Additional, these devices are often used for remote control of IoT devices or for convenient access to IoT device interfaces. In contrast to devices in the class "IT", smartphones and tablets usually do not offer services to other devices (e.g., no SSH server or media server running on smartphones).
Communication	IP-telephone, fax	Protocols in use are limited to typical protocols for voice-over-IP-communication.
Server Mode Devices	Devices that open a server (e.g., IP-Cameras)	Devices offering services to other devices in the network/Internet. Typically open ports to the Internet.
Home Entertainment	Game console, HiFi system, Smart TV	Typically communicate with entertainment companies (e.g., provider of online games). May be the source of considerable amount of traffic.
Storage and Streaming	Smart TV, NAS	Communicate with streaming services or cloud storage. May causes considerable amount of traffic
Energy	Heater, air condition	Important devices, since they have an influence on well-being of users. Usually do not generate much traffic. May communicate with energy provider (smart grid) or other energy-related services in the Internet.
Safety	Smoke detector, door	Critical devices, since they have an influence on human safety. Usually only communicate in the local network.
Health	Smart toothbrush, smart glucose meter	Class may include some critical devices, since they have an influence on human safety. Only have limited communication to the Internet
Comfort	Bed mattress, massage chair	Rather unimportant devices, no Internet communication.
Household	Fridge, washing machine	Important for everyday life, little Internet communication (e.g., for smart grid purpose to supervise/ control energy usage)
Automation	Devices for automation like "Homee"	Must communicate with a lot of different devices, but limited to communication in the home network.
Gadgets	All kind of gadgets like alarm clock, weather sta-	Difficult to describe the traffic, because many devices with different tasks belong to this class. However, these devices often

	tion	have very limited Internet communication (e.g., only with a weather service).
Toys	Teddy bears, remote controlled car	Rather unimportant devices, usually only with limited Internet communication. No communication with other classes necessary.
Other Devices	Other devices	Difficult to describe the traffic, because many devices with different tasks belong to this class. This class should have strict security policies.

Gateways between legacy/proprietary networks may exist. Gateways are typically used to integrate legacy/proprietary networks into the smart home WiFi network. The smart home security system on the router can not operate inside legacy or proprietary networks, but it can affect the traffic which goes inside and outside the network and passes the router. Gateways get assigned the class that best describes the devices in the legacy/proprietary network.

B. Hierarchical Ordering

The approach presented in this paper asks for the most precise possible description of data traffic and communication partners to be useful. If the description of data traffic is too general, the smart home security system cannot effectively restrict the communication or it erroneously allows traffic. If the description of traffic is too strict, it becomes too complex or would increase the false alarm rate (especially false negatives). As already mentioned, it is hard to describe the set of allowed communication partners for each device. Therefore, a hierarchical ordering is helpful. This ordering enables making decisions on a more abstract level. That means it is possible to state that a device cannot communicate with a device of a special class (e.g., a special toy is not allowed to communicate with household devices, or even that toys can't communicate with health devices at all). The smart home security system uses six hierarchical levels, shown in Table 2.

TABLE II. HIERARCHICAL ORDERING IN A SMART HOME

Level	Name	Description	Categorization	Configured by
6.	**Environment**	Environment	Network	System
5.	**Subnet**	Subnet		
4.	**Class**	Class of device	Classification	
3.	**Type**	Type of device		
2.	**Union**	Union of devices	Device	Manufacturer
1.	**Device**	Single Device		

This ordering allows defining security policies on different levels, e.g.,

- for the whole home network (level 6),
- for a subnet (level 5),
- for different device classes (level 4), for the list of classes (see Table 1),
- types of devices (level 3) like Smart TV or Heater,
- a union of devices (level 2) that make it possible to set up rules for devices of the same manufacturer or same subsystem,
- and a single device itself (level 1).

The levels fall in one of three categories:

- Network (level 5 and 6),
- classes (level 3 and 4), and
- device (level 1 and 2).

Security policies for the levels "network" and "classes" are preconfigured on the home router. These rules originate from the company implementing the smart home security system for the home router and may be extended by third parties, or the owner of the home router.

The "Device" rules originate either from the device itself, from trusted third parties, from a profiling algorithm, or from the user. See Section IV.C for a more detailed description. Table 3 shows an example of the use of the hierarchical levels.

TABLE III. SMART HOME HIERARCHY EXAMPLE

6	**Environment**	Smart Home							…	
5	**Subnet**	Subnet 1						…		
4	**Class**	Energy					…			
3	**Type**	Heater				…				
2	**Union**	Company 1		Company 2		…				
1	**Device**	Heater 1	Heater 2	Heater 3	…					

Manufacturers of IoT devices are only allowed to influence security policies on the device levels "union" and "device", and a single device may only influence security policies regarding itself. Hence, a device may define communication from itself to another device, from itself to the network, from the network to itself, from the device to a class of devices and from a class of devices to the device.

Security policies from higher levels overrule security policies at lower levels. That means if a manufacturer of a toy wants to allow communication from the toy to a health device but the communication between toys and health devices is forbidden on the class level, the home router forbids this communication. Security polices should follow the security principle "least privilege". That means that the scope of the permissions of devices should be as limited as much as possible.

C. Integration Process

All relevant security processes take place when a device joins the network. In most home networks, the Dynamic Host Configuration Protocol (DHCP) [14] is used to dynamically assign IP addresses to devices and send additional configuration data. The smart home security system presented in this paper piggybacks on DHCP. The DHCP pro-

 ISBN: 978-1-61208-582-1

tocol is executed at every initiation of a device. An ideal sequence, without disturbances, is shown in Figure 2. The home router acts as DHCP-Server. This section gives an overview on different methods for device integration.

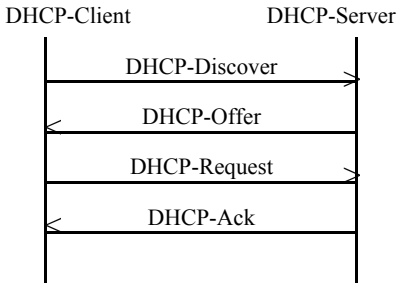

Figure 2. Typical DHCP sequence

1) Integration using Self-descriptions

The self-description approach requires the device's manufacturer to be aware of the system described in this paper. In a nutshell, the device provides a self-description of the intended communication partners of the device as well as a detailed description of traffic patterns produced by this device. Self-descriptions come in the form of attribute certificates and are signed by the device manufacturer. The integration of the device using self-description is nearly fully automatic. In fact, the user is only involved once to ask if a device should really get integrated into the network. This results in good usability of the integration process. The home router company may decide to allow for additional configuration using the home router administrative user interface (e.g., a web application running on the home router).

When a device sends a DHCP-Discovery, the home router takes notice of this device. In the DHCP-Offer, it starts the integration process. When a device gets connected to the home network, it transmits its identification data first, including a firmware version. The following situations can occur:

- A: Device known to home router, firmware version known to home router, signature of self-description valid
- B: Device known to home router, firmware version known to home router, signature of self-description invalid (e.g., signing key no longer valid)
- C: Device known to home router, firmware version not known to home router
- D: Device unknown to home router

In the case of situation A, the home router continues the DHCP protocol and integrates the device into the network. All security policies are enforced. In the case of situations B and C, the home router requests the self-description again. Only self-descriptions from the same manufacturer are accepted and only for the same type of device. The home router validates the possession of the private key associated with the self-description (attribute certificate) to make sure it has been issued for this device. By doing so, the home router ensures that a hacked device could not gain more communication privileges by reusing the self-description of another manufacturer or the self-description of the same manufacturer but for another kind of device. After the integration, all updated security policies are enforced. In the case of situation D, the home router also requests the self-description but self-descriptions of all manufacturers are accepted. As described above, the system presented in this paper assumes a device that is integrated in the smart home network for the first time is to be trusted ("leap of faith"). However, to avoid a hacked device using the self-description of a device from another manufacturer or another device class, the user is queried to confirm that a new device was added to the home network. In all cases, after a successful transmission of the self-descriptions, the allowed communication partners as well as the traffic characteristics are stored in the routers database together with the device identification, device credentials for secure IDs, and the firmware version. Security policies are updated according to the new information and all security policies get enforced.

2) Integration using the built-in scanner

It is very likely that the smart home security system presented in this paper will need an extended period of time to become adapted by all smart home device manufacturers (if ever). The scanner described in this section allows for support of legacy devices as well as support of devices by manufacturers that willingly decide not to support this system. The scanner profiles devices, identifies them, and acquires an appropriate description of allowed communication partners and communication characteristics from trusted third parties. Such trusted third parties are quite common in other security domains, e.g., web filtering or spam detection. If the system cannot obtain the necessary description of a device, manual integration by the user is necessary. The scanner is invoked during the DHCP-protocol if the home router does not receive any self-description of the device. In this case, the user is queried if there really is a new device in the network to prevent an attacker from hacking a device and then trying to trick the scanner to identify the hacked device as a different device than it is. If the user confirmed that there is a new device, the scanning process starts. The home router uses methods from penetration testing to identify characteristics of the device, e.g., it scans for open ports, grabs banners of available services, fingerprints TCP/IP communication, etc. All the resulting characteristics are uploaded to the trusted third party that compares those characteristics to its database of known IoT devices. The third party returns the security policy to apply. If the fingerprinting does not work, the user can select the device with the app via a given list or it would also be imaginable that he is scanning the product code from the packing of the device. If it is successful, the scanner tries to download the identification and communication data from an external data source (manufacturer or trusted third party).

3) Manual integration

The third option is the manual integration of the device by the user via a smartphone app. There are four different ways to do so. W1 is analogous to the scanners alternative, if the fingerprinting does not work. W3 and W4 do not need

a traffic profile to integrate the device into the home network.

- W1: The user is asked to enter the type of device, manufacturer, and model. Alternatively, the user scans the product code from the packaging of the device. All associated data is retrieved from a trusted third party, which returns the security policy to apply.
- W2: The user downloads the identification data and communication data manually from the manufacturer's website or trusted third party and imports it.
- W3: The user enters only the type of device and accepts the generic security policy for this type (level 3 in the hierarchy model).
- W4: The user enters the allowed communication partners as well as communication characteristics by hand. It is highly recommended to avoid this approach, as it is error prone.

V. IMPLEMENTATION

The security system for smart homes is currently getting implemented on a standard home router (TP-Link TL-WR841ND) using a Linux distribution for home routers (OpenWRT version Chaos Calmer v15.05.1). The current implementation is a proof-of-concept subset of the security system described in this paper: it solely uses integration by self-description and a feature limited version of traffic descriptors (basically rules for packet-filter firewalls). User interaction uses the administration interface of the home router. Challenges for implementation of the complete security systems for smart homes include handling the complexity of the full syntax traffic descriptors, certificate handling, efficient handling of security policies in the hierarchical model, and reducing memory usage and performance overhead. A major challenge will be an efficient implementation of the scanner for the integration of legacy devices. The scanner will be part of future research, as it also requires more conceptual work.

VI. CONCLUSION

This paper presented a smart home security system with a special focus on IoT devices in smart homes. The smart home security system enforces security policies per class of IoT devices. Such security policy limits the communication of IoT devices to a predefined set of communication partners, and hence protects the Internet from hacked IoT devices. IoT devices from different classes are isolated such that a security incident in one class of devices cannot influence the other devices, thereby limiting the outbreak of an attack. If IoT devices support the smart home security system presented in this paper, only one user interaction is necessary during integration of new devices. There is also a process to integrate legacy devices that requires slightly more user interaction. The proposed security system offers full transparency of communication partners of IoT devices during their integration into the network. This transparency enables consumers to buy only IoT devices that satisfy their security and privacy needs (e.g., by buying only IoT devices communicating with communication partners in countries implementing the General Data Protection Regulation).

REFERENCES

[1] B.Krebs, "DDoS on Dyn Impacts Twitter, Spotify, Reddit", in: "Krebs on Security – In-depth security news and investigation", https://krebsonsecurity.com/2016/10/ddos-on-dyn-impacts-twitter-spotify-reddit/ , October 2016 [accessed 05/23/2017].

[2] Hewlett-Packard Development, "Internet of Things Resarch Study", September 2014.

[3] L. Braun and H.-J. Hof, "Smart Home Security", Poster, Applied Research Conference 2016, Augsburg, Germany, 2016.

[4] H.-J. Hof, "Towards Enhanced Usability of IT Security Mechanisms – How to Design Usable IT Security Mechanisms Using the Example of Email Encryption", International Journal On Advances in Security, volume 6, number 1&2, pp. 78-87 ISSN 1942-2636, 2013.

[5] H.-J. Hof, "User-Centric IT Security – How to Design Usable Security Mechanisms" , The Fifth International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC 2012), pp. 7-12, November 2012.

[6] H.-J. Hof and G. Socher, **"**Security Design Patterns with Good Usability"**,** 9th ACM Conference on Security and Privacy in Wireless and Mobile Networks (ACM WiSec 2016), Darmstadt, Germany, pp. 227-228, July 2016.

[7] S. Fries and H.-J. Hof, "Secure Remote Access to Home Energy Appliances" in: Lars Torsten Berger, Krzysztof Iniewski, "Smart Grid: Applications, Communications, and Security", John Wiley & Sons, Inc, pp. 443-454, ISBN: 978-1-118-00439-5, 2012.

[8] R. Falk, S. Fries, and H.-J. Hof, "ASIA: An Access Control, Session Invocation and Authorization Architecture for Home Energy Appliances in Smart Energy Grid Environments", in The First International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies (ENERGY 2011), pp. 19-26, ISBN: 978-1-61208-136-6, Mai 2011.

[9] C. Müller, J. Schmutzler, C. Wietfeld, S. Fries, A. Heidenreich, and H.-J. Hof, "ICT Reference Architecture Design based on Requirements for Future Energy Grids", First International Conference on Smart Grid Communications (IEEE SmartGridComm 2010), pp. 315-320, ISBN: 978-1-4244-6510-1, Oktober 2010.

[10] N. Komninos, E. Philippou, and A. Pitsillides, "Survey in Smart Grid and Smart Home Security: Issues, Challenges, and Countermeasures", IEEE Communications Surveys&Tutorials, Vol. 16, No. 4, pp. 1933-1954, 4/2014.

[11] D. Pishva and K. Takeda, "Product-Based Security Model for Smart Home Appliances", IEEE A&E Systems Magazine, pp. 32-41, 10/2008.

[12] B. Schneier, "Lessons From the Dyn DDoS Attack", in "Schneier on Security", https://www.schneier.com/blog/archives/2016/11/lessons_from_th_5.html, November 2016 [accessed 05/23/2017].

[13] Y.Xu, Y. Jiang, C. Hu, L. He, and Y. Cao, "A balanced security protocol of Wireless Sensor Network for Smart Home", Proceedings of 2014 IEEE 12th International Conference on Signal Processing (ICSP), HangZhou, China, pp. 2324-2327, 2014.

[14] R. Droms, "Dynamic Host Configuration Protocol", RFC 2131, https://tools.ietf.org/html/rfc2131, March 1997 [accessed 08/31/2017].

 ISBN: 978-1-61208-582-1