Securing Smart Homes using Intrusion Detection Systems

Christoph Haar

Hochschule für Telekommunikation
Gustav-Freytag-Straße 43-45
Leipzig, Germany
Email: haar@hft-leipzig.de

Erik Buchmann

Hochschule für Telekommunikation
Gustav-Freytag-Straße 43-45
Leipzig, Germany
Email: buchmann@hft-leipzig.de

***Abstract*—Botnets, such as Mirai or Reaper show that many Smart Home devices are low-hanging fruits for attackers. Nevertheless, it is an ongoing trend to replace everyday devices, such as TV, fridges or doorbells by smart successors. Thus, securing Smart Homes operated by private users remains an open issue. In this paper, we explore options to integrate an Intrusion Detection System (IDS) in a Smart Home installation. Smart Home devices use well-established technology. From a technical perspective, existing IDS approaches can be applied. We focus on non-technical challenges. This includes a system design that allows for a pre-configuration. It also calls for processes which allow users to invoke a security expert in the case of an attack that cannot be handled by simple means. We demonstrate our approach with a prototypical implementation.**

***Index Terms*—IT Security; Smart Home Security; Intrusion Detection Systems**

I. Introduction

The number of Smart Home devices is increasing day by day. Almost any recent TV is "‘smart"’. It possesses computational resources, an operating system, various applications, and an Internet connection via WLAN. Countless everyday devices from lightbulbs [1] to gardening equipment [2] have smart successors that use the Internet to provide new modes of use. Typically, Smart Homes are operated by private users without IT-Security expertise. For such users it is not obvious that the new TV needs frequent security updates, while its non-smart predecessor could be used years without care. It is also not obvious that security measures like a simple firewall on the Internet router or an anti-virus software on some devices cannot protect the Smart Home sufficiently. Botnets, such as Mirai [3] or Reaper [4] show that Smart Home devices are in the focus of adversaries already.

A well-established approach to ward off such risks is to use an Intrusion Detection System (IDS) [5] [6]. An IDS detects attempts to break into a network segment and allows the user to take appropriate countermeasures. From a technical perspective, existing IDS consider network protocols, services, operating systems, software libraries, etc. that are used by Smart Home devices. However, due to some non-technical aspects it is challenging to apply IDS to Smart Homes:

It is not feasible for a private user without security expertise to configure an IDS. It is neither feasible for this user to distinguish between a false alarm and an attack, and to identify appropriate countermeasures. Furthermore, it must be explainable to the private user in which way an IDS secures a Smart Home installation, which devices are secured, and who is responsible to what extent if an attack goes unnoticed. It is also problematic to integrate an IDS into a Smart Home as a security appliance, which is constantly configured, monitored and maintained by an external security expert. First, this approach is prohibitively expensive for private users. Second, the security expert would have full access to the monitored network segment, which violates the privacy of the user.

In this paper, we focus on two research questions:

1) How can an IDS be integrated into a Smart Home operated by private users without IT-Security expertise?
2) Which IDS approaches can be adapted for that purpose?

We systematically explore how network segmentation, system architecture, security process and specification of product features for an IDS must be adapted to secure Smart Home installations. By means of an experiment, we demonstrate that both anomaly-detecting IDS and signature-detecting IDS are applicable, but the latter ones generate fewer false alarms.

Paper structure: In Section II, we review related work. In Section III, we provide a problem statement. We will answer the first research question in section IV and the second research question in Section V. Section VI concludes.

II. Related Work

In this section, we briefly describe Smart Homes, existing IDS approaches and components, and the IT-Security Process.

A. Smart Homes

The term “Smart Home” refers to the use of information and communication technology for domestic use [7]. This ranges from (a) home automation over (b) controlling domestic appliances to (c) smart devices with extended modes. An example for (a) is the use of smart gardening equipment [2] that waters the plants depending on the weather and moves the lawn automatically. An example for (b) is a smart light bulb [1], which simulates an indoor sunset and synchronizes with a movie shown in TV. Finally, an example for (c) is a smart speaker [8]. By using a cloud service to realize voice control, a smart speaker plays music, reads emails and news, manages appointments etc. The sum of all smart devices is a Smart Home installation. Since a Smart Home connects devices in private spaces to the Internet, it is problematic both from a privacy and security perspective [9].

B. Intrusion Detection Systems

IDS strive to detect attacks [10] to the devices in the network. Such attacks might come from the outside, e.g., over the Internet. Insiders are also possible sources of attacks, e.g., employees. Typical attacks include *Scanning Attacks* like Portscans or network scans [11]. Scanning attacks help an attacker to identify potential vulnerabilities in a system. *Denial of Service Attacks* flood a network or a device with data packets. Since such packets consume computational resources, the availability of the attacked system is at stake [12]. Service-specific attacks, such as a *Telnet Attack* aim for vulnerable services [13]. Recently, many Smart Home devices allowed unencrypted access with a hard-coded administrator password, which can be exploited with a Telnet Attack.

Host-based IDS detect attacks directly at the monitored devices [14]. To implement a host-based IDS, it must be possible to install software on the devices that should be secured. In contrast, *Network-based IDS* are stand-alone systems that monitor entire network segments [10]. For this purpose, in each segment a network appliance, such as a router, bridge or firewall must sent a copy of all data packets to the IDS. This allows to secure all devices in a network segment without having to install software on each device.

C. IDS Components

Each IDS realizes a number of components. A *Knowledge Base* contains all information necessary to distinguish an attack from normal network traffic. Information about the current state of the IDS is provided by a *Configuration Component*. A *Sensor* fetches data packets gathered at an *Information Source*, i.e., a monitored device or an appliance in a certain network segment. The *Detector-ID Engine* compares the data from the Sensor with the information from the Knowledge Base to identify attacks. If an attack is detected, a *Response Component* raises an alarm and initiates an automated or an human involved action [15].

Two alternatives exist to implement the Detector-ID Engine. A *signature-detecting IDS* applies a preconfigured set of pattern and rules (the signature) to the data packets in order to identify attacks. These signatures can be defined by the IDS operator according to match a company-wide IT-Security policy. It is also possible to import signatures from well-researched attacks from external repositories [16].

Anomaly-detecting IDS use machine learning and artificial intelligence to learn what is normal data traffic [17]. A voting algorithm decides if new data packets differ so much from normal data traffic that an alarm is generated.

Typically, an IDS comes with a basic pre-configuration that considers the characteristics of the implemented components. However, this pre-configuration is only meant to speed up the configuration process for the security expert, and to demonstrate the use of configuration parameters. Using an IDS out of the box does not result in a reasonable network-security advantage. Thus, existing IDS approaches must be part of an IT-Security Process, which is executed by security experts [18], [19], [20].

D. IT-Security Process

The IT-Security Process follows a plan-do-check-act cycle [21]. In the *Plan* phase, the management defines a general IT-Security policy. Furthermore the needed controls and procedures are identified. In the *Do* phase the identified controls and procedures are implemented. During the *Check* phase all the implemented controls and procedures are evaluated. In this phase security incidents are identified as well. The *Act* phase includes a constant improvement of the implemented measures based on the identified security incidents. These improvements are leading back to Plan, in which the policy can be improved [22]. Depending on the company structure, different persons are involved in this process. However, every person needs expertise in IT-Security.

The phases of the generic IT-Security Process are adapted to the needs of an IDS, as follows: In the *Plan* phase, the IDS is configured to distinguish attacks from normal network traffic. In the *Do* phase, those information are implemented in an IDS instance. In the *Check* phase, the IDS detects attacks. Finally, in the *Act* phase the performance of the IDS is reviewed to adapt the Knowledge Base for attacks that went unnoticed.

III. Problem Statement

We strive to integrate an IDS in Smart Home installations connected to the Internet. To this end, we distinguish two roles:

A ***security expert*** possesses the IT-Security expertise needed to develop an IT-Security policy, to configure an IDS respectively, to operate the IDS and understand its alarms, and to react with appropriate measures to alarms.

A ***private user*** lacks this kind of expertise. Such a private user can follow manuals written without technical vocabulary. It is difficult for a private user to find out if an IDS alarm comes from an attack or a misconfigured network appliance.

Our objective is to use an IDS to increase the security of a Smart Home installation in the possession of a private user.

Observations show that Smart Home devices use protocols, libraries and technologies which have been developed for years [23]. From a technical point of view it is feasible to configure an IDS [24] for Smart Homes. However, IDS approaches have been developed to secure complex corporate networks. Existing IDS put an emphasis on the integration into security management processes, which allow experts to implement a comprehensive security strategy. It is not in the focus of such IDS to provide intuitive explanations.

In order to integrate an IDS into a Smart Home, we specifically consider non-technical aspects of an IDS. Our starting point is a set of three requirements that arise from security challenges for Smart Home devices:

Expertise: The user does not need to possess in-depth expertise of technical internals, such as network protocols and IT-Security [25]. This requirement is valid for any Smart Home device tailored for private users.

Separation: Smart Home devices have dedicated use cases that can be separated from others. In many cases, Smart Home devices have traditional, non-smart predecessors. Such predecessors have built expectations and experiences regarding modes use and handling [26] [25].

Understandability: The interaction between a user and a Smart Home device should be as understandable as possible [27], [28]. This is challenging, as private users cannot expected to comprehend technical vocabulary.

IV. An IDS approach for Smart Homes

To systematically approach an IDS that secures Smart Homes, we investigate the four levels *Network Segmentation*, *System Architecture*, *IT-Security Process* and *Contract Liabilities*. Our levels have been compiled from proposals to secure Smart Home networks [6], [25], from well-known IT-Security concepts [18]–[21], and from challenges discussed in the IDS context [15], [16], [24]. In the following, we briefly explain each level, and we apply the requirements from Section III.

A. *Level* Network Segmentation

The concern of this level is to separate the Smart Home devices under observation of the IDS from all other devices that might be part of the network of the user.

Existing IDS approaches are configurable for corporate networks. Such networks feature multiple segments which transport data from different applications. Each segment comes with specific security requirements. Within each network segment, a network appliance, such as a router or a firewall sends copies of the data stream to an IDS, e.g., via a Security Incident and Event Management System. Alternatively, IDS software components can be installed on each device in a network segment (cf. Section II-C). However, typical Smart Home installations use a simpler configuration, as shown in Figure 1. In the figure, arrows describe data transmissions and rounded rectangles depict network segments.

From Requirement **Separation** follows that it must be clear which devices are under observation. We propose to span a separate Smart Home network containing all Smart Home devices, as illustrated in Figure 2. All devices in the Smart Home network have similar properties and security requirements. That is, the Smart Home devices have a single purpose, observe the user context, handle person-related data and possibly communicate over the Internet. For this reason, it makes sense to operate all Smart Home devices in a separate network. Furthermore, in case of an attack, conventional devices such as PCs or laptops remain unaffected.

Requirement **Expertise** rules out host-based IDS that require a technically demanding installation and configuration. Figure 1 shows that the best place for an IDS in a Smart Home installation is the router. The router controls the network boundaries and handles data transfers between the Smart Home devices. Figure 2 illustrates this approach.

Regarding **Understandability**, an isolated network for Smart Home devices allows to explain to the private user which devices are under observation and where security alarms are located. Because all devices in the Smart Home network have similar security properties, it is not necessary to let the private user generate a complex IDS configuration. Instead, the IDS can be preconfigured for typical Smart Homes.

B. *Level* System Architecture

This level considers the system architecture of the IDS. Figure 3 depicts a typical IDS installation (cf. Section II-B). Components are depicted as gray rectangles, black lines illustrate information flows and ovals represent roles. The dashed lines are responsibilities. All components that need supervision or configuration are assigned to the security expert. This is particularly problematic for the Response Component. It delivers alarms which can be explained only when knowing the signatures that have been configured. We tackle this issue by modifying the information flows, changing responsibilities and introducing a new component, as shown in Figure 4.

Separation calls for a clear distinction between different tasks. We distinguish between a *preconfiguration stage* and an *operational stage*. The components assigned to the preconfiguration stage are in the responsibility of the IDS manufacturer. In particular, the manufacturer possesses security experts, which specify the general IT-Security policy and signatures for the Knowledge Base. The components assigned to the operational stage are in the responsibility of the private user.

Expertise means that the private user cannot be expected to take actions depending on expert knowledge. With our approach, the components in the operational stage are automated so that no expert knowledge is necessary.

However, the Response Component cannot be fully automated. The response to an alarm depends on the Smart Home devices installed, the kind of alarm and the IDS (pre)configuration, which violates **Understandability**. To solve this issue, we introduce a Reporting Component. This component allows to invoke a security expert with all information needed to find out if it was a false alarm, and to devise an adequate response if not. In particular, the Reporting Component automatically generates a report, based on the system state from the Configuration Component and the alarms from the Response Component.

Observe that the Reporting Component forwards reports to a security expert only if instructed by the user. Thus, the security expert cannot permanently observe the Smart Home network. Because Smart Homes typically cover private areas of the user's life, this is important.

C. *Level* IT-Security Process

The level IT-Security Process ensures that there is an appropriate response on IDS alarms, and the IDS will be adapted to changing properties of the network if necessary.

Corporate networks are frequently adapted to new demands, and adversaries might develop new attacks. An IDS increases the network security only if it is constantly monitored and improved. To this end, an IDS is part of the company's IT-Security Process, as shown in Figure 5. In the figure, rectangles denote process steps and black lines the information flow. Ovals depict roles and dashed lines responsibilities.

With our approach, only Smart Home devices are part of the Smart Home network monitored by the IDS. Such Smart Home devices rarely change its functionality. The security

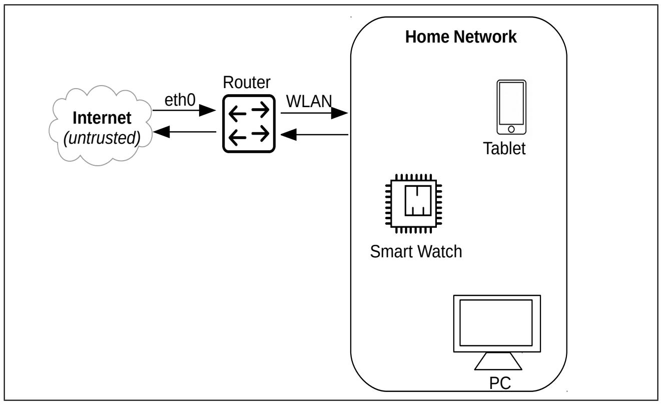

Fig. 1. Typical Smart Home Architecture

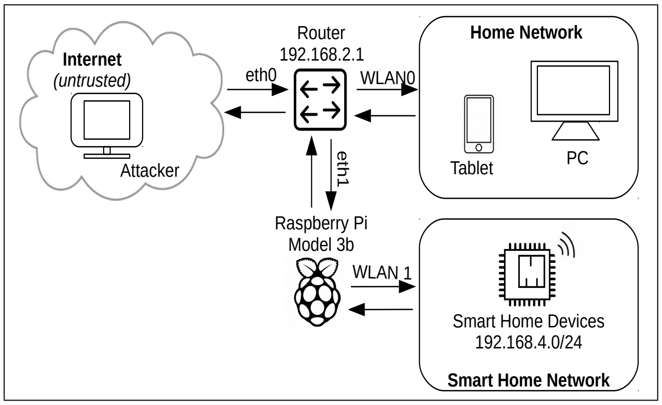

Fig. 2. Experimental Smart Home Architecture

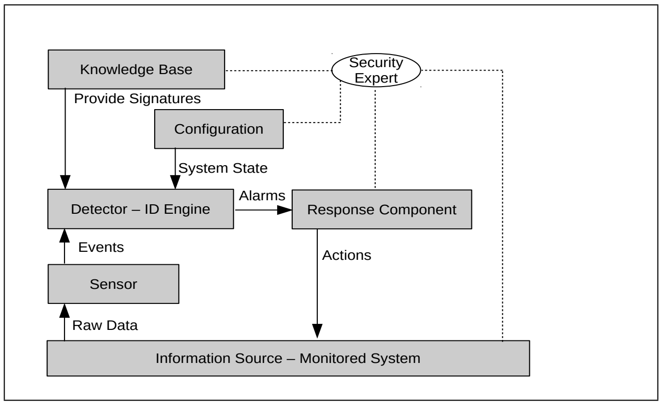

Fig. 3. Existing IDS

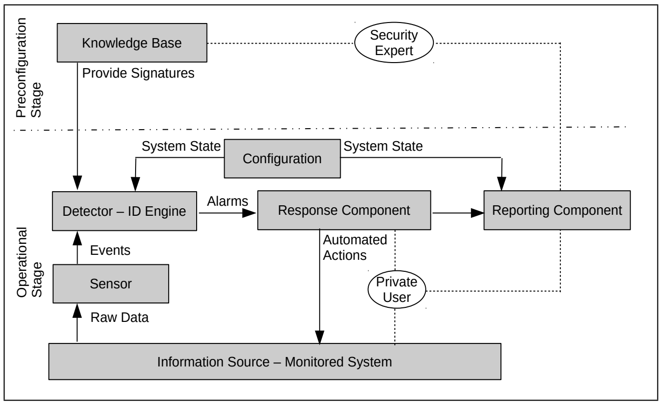

Fig. 4. Smart Home IDS

requirements of the Smart Home network stem from the Smart Home concept and do not change over time. Thus, the IT-Security Process can be streamlined for Smart Homes.

Separation requires to separate the IT-Security Process into phases that have a distinct purpose. We distinguish between four phases *Preconfiguration*, *Installation*, *Detection* and *Countermeasures*, as shown in Figure 6.

The Pre-configuration phase is related to the pre-configuration stage from Section IV-B. Regarding **Expertise**, a private user cannot be expected to devise an IDS configuration. With our approach, the security expert defines an IDS configuration tailored for a network segment with the security properties of a Smart Home network.

The other phases are taking place at the operational stage, i.e., make use of IDS components that have been automated. In the Installation phase, the private user must only connect the IDS to the Internet router and the Smart Home devices to the IDS. After that the IDS starts monitoring the Smart Home network. In the Detection phase, the IDS automatically identifies potential attacks and raises the alarms if necessary.

In the Countermeasures phase, the IDS suggests actions to the private user to ward off attacks. If the IDS detects an attack that has been preconfigured in the Knowledge Base, it suggests reasonable measures, e.g., re-starting or disconnecting the Smart Home device. If there is no countermeasure that is explainable to the private user, **Understandability** means that the private user must invoke an external security expert. In this case, the Reporting Component helps the user to provide the security expert with all information necessary to devise reasonable measures, and to update the Knowledge Base.

D. Level Contract Liabilities

This level considers in which product features a Smart Home IDS manufacturer can assure to a private user.

Traditional IDS are sold as a "‘construction kit"’, which needs to be configured by the customer’s security expert to be effective. If such an IDS does not ward off an attack, it is in the responsibility of the security expert. The expert has compared the abilities of the IDS with the demand of the company network and generated the configuration of the IDS. However, such an approach is not suitable for private users.

From **Separation** follows that a Smart Home IDS must be able to define a distinct service. With our approach, the manufacturer can define this service in the pre-configuration phase. It includes all devices in the Smart Home network that are connected to the IDS.

Expertise requires to specify the abilities of the IDS without referring to certain transmission protocols or attack names. However, many Smart Home devices have a similar architecture and use similar communication protocols [29]. Thus, it

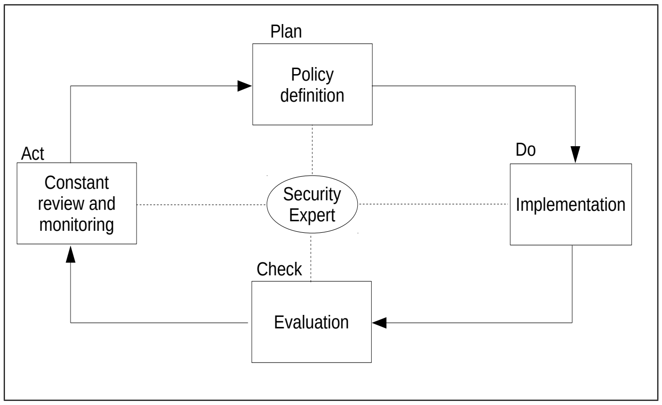

Fig. 5. IT-Security Process

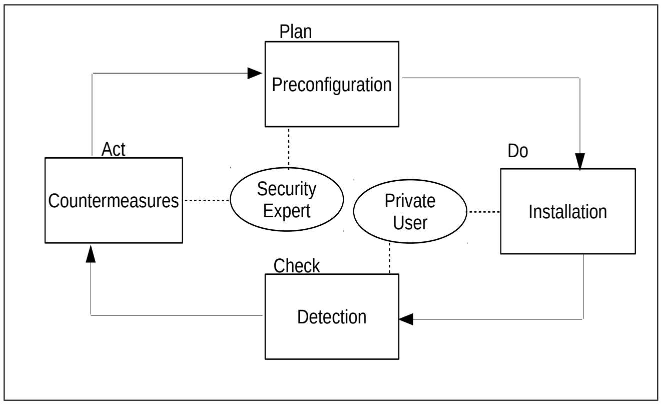

Fig. 6. Adapted IT-Security Process

might be feasible to promise a certain level of protection for certain product groups or manufacturers.

Requirement **Understandability** means that it must be clearly communicated to the private user that an IDS does not offer a complete protection against any kinds of attack to the Smart Home devices. Otherwise, the manufacturer would be held responsible in case of an attack.

E. Discussion

We have shown how an IDS can be used to secure a Smart Home operated by private users, on four distinct levels. However, this might result in new challenges. One issue is that the IDS is operated in a potentially insecure environment. For example, a private user might turn off the IDS by mistake, or misconfigure the Internet connection so that the Reporting Component cannot send information to the security expert.

Another potential issue is the check phase of the IT-Security Process, which we have automated. If mistakes in the preconfiguration result in successful attacks, such attacks might not be apparent during the check phase. Thus, the IDS will not be updated for this kind of attack.

V. Signature- or Anomaly-Based Detection

In this section, we want to confirm that current IDS can be used as described in Section IV. We also want to find out if signature-based or anomaly-based IDS are better suited.

A. Experimental Setup

We have conducted experiments with the system architecture illustrated in Figure 2. The IDS was installed on a Raspberry Pi 3B that operates as a Wi-Fi Bridge between the Smart Home network (WLAN1) and the Internet router (eth1). The Raspberry Pi 3B is sufficient to evaluate network packets in real-time. We have tested two different IDS:

Suricata realizes a signature-based detection. Reviews [30] show that Suricata is widely used, implements state-of-the-art detection algorithms and makes use of multi-core processors. Suricata starts with approx. 27.000 preconfigured signatures, and it allows to update the signatures from a repository.

Kitsune is an anomaly-detecting IDS which implements a number of neuronal networks to detect attacks [31]. To this end, Kitsune constructs a feature vector from each data packet, which is transferred to the set of neural networks. The output of the networks is forwarded to a voting mechanism. Kitsune is installed with neuronal networks and a voting mechanism that are pre-trained and preconfigured for services and network protocols that are also used by Smart Home devices.

Both IDS approaches provide the features needed according to Section IV: Both approaches are network-based IDS, which can be installed on a bridge between Internet and Smart Home network. Suricata and Kitsune use a modular architecture, which allows to implement the components shown in Figure 4. Finally, both IDS can be preconfigured and updated remotely by a security expert, which is needed by our security process.

Our Smart Home network contains four different devices:

- The *Amazon Dash-Button* connects to the Smart Home network when the button is pressed. Then it fetches the current time from an NTP server over the Internet, opens a HTTPS connection to the Amazon cloud and places an order for a specific product. After that, it disconnects from the network until the button is pressed again.
- The *Amazon Echo Dot* (2nd generation) is a smart speaker with a voice assistant. As soon as the speaker recognizes a wake-up word, it sends voice samples to the Amazon cloud for natural language processing. The response data that is sent to the smart speaker depends on the voice command. A wide number of activities from playing music to controlling other Smart Home devices is supported.
- The *Temperature Sensor* communicates via MQTT protocol [32] with a server that logs temperature readings. To this end, the Temperature Sensor resolves an IP address for a preconfigured domain from an DNS, and connects to this IP at port 1883.
- The *IP-Camera* is always connected to a server with the IP address 35.177.224.169. This server is used to establish connections between a client and the IP-Camera. Thus, the private user can connect to the IP-Camera from different networks.

TABLE I. STAGE 1: NORMAL USE

Device	Intervall	Duration	Interactions
Amazon Dash	10 minutes	1 sec.	6
Amazon Echo	10 minutes	5 minutes	6
IP-Camera	10 minutes	2 minutes	5
Temperature	10 seconds	-	60

B. Experimental Procedure

Our experiment takes place in three stages. In each stage, the Raspberry Pi records all data packets using tcpdump.

In the *first stage*, all four Smart Home devices were used for 60 minutes. Table I shows in which time intervals and for how long each device was used. For example, the first line means that the Dash Button was pressed every 10 minutes for one second. The Temperature Sensor sends the temperature automatically every 10 seconds. In this stage, we have recorded 112.602 packets. All packets refer to normal operations.

In the *second stage*, we have used nmap to perform a Portscan from the Internet to the Smart Home network. With a Portscan, a network appliance will be searched for ports that are open to the Smart Home network. A Portscan is a threat, because it identifies characteristics of a device. This includes the services it offers to the network, and the applications or software libraries listening to open ports. Our Portscan starts after 48 minutes of normal activity. In total, we have recorded 237.609 packets, and 131.137 of them belong to the attack.

In the *third stage*, we have executed a Telnet Attack from the Internet to the Smart Home network. Telnet is a plain-text protocol to access devices offering unencrypted services. For example, the Mirai Botnet used a Telnet Attack to infect Smart Home devices with a hard-coded admin password. To mimic a Telnet attack that was successful, we have extended the firmware of the Temperature Sensor with a simple Telnet server. Again, the attack starts after 48 minutes. We have recorded 114.501 packets, 1.107 of them belong to the attack.

After the execution of the stages, Suricata and Kitsune process the records. Thus, both IDS analyze the same data.

C. Experimental Results

In this section, we evaluate if the IDS approaches are (a) sufficiently accurate to increase the security of a Smart Home installation and (b) applicable for a private user. Regarding (a), we map the detection results to a confusion matrix. Such a matrix shows in each column the number of packets the IDS has classified as malicious or benign. Each row shows which packets were indeed malicious or benign. With an ideal IDS, only the upper left and lower right fields in the matrix contain numbers > 0.

1) Normal Operations: As Table II shows, Suricata identified all packets correctly as begin. In contrast, Kitsune has misclassified 43 packets as malicious.

TABLE II. NORMAL OPERATION

		Suricata		Kitsune	
		Malicious	Benign	Malicious	Benign
Reality	Malicious	0	0	0	0
	Benign	0	112.602	43	112.559

2) Portscan Attack: Table III contains the classification of the packets from the Portscan. A Portscan might have a benign reason. For example, a network operator might want to confirm that all network services are well. On the other hand, a Portscan can be the first step of an attacker who wants to identify vulnerable services. Suricata has identified 48 packets from the Portscan as malicious, but 131.089 others as benign. During our experiments, we have learned that Sucicata does not consider a Portscan as an attack. Thus, depending on the point of view, either 48 or 131.089 packets were misclassified.

In contrast, Kitsune has classified 129.987 packets from the Portscan as malicious. Sending packets to all ports on all Smart Home devices differs from normal network operations. Kitsune recognizes this behavior as an anomaly and raises an alarm.

TABLE III. PORTSCAN

		Suricata		Kitsune	
		Malicious	Benign	Malicious	Benign
Reality	Malicious	48	131.089	129.987	1.150
	Benign	0	106.472	178	106.294

3) Telnet Attack: No device must allow unencrypted login over the Internet. Thus, a Telnet access is an attack. As Table IV shows, Suricata has correctly identified all benign and malicious packets. To our surprise, Kitsune was unable to identify malicious packets. We think that this is because the unencrypted TCP packets sent by the Temperature Sensor with low data rate resemble the Telnet packets from our attack. Furthermore, Kitsune has classified 2.848 benign packets as malicious. In this case, we observed that Kitsune was confused by the user switching the radio station played by the Echo Dot. This caused an anomaly in the data transfers, but must be considered a false alarm.

TABLE IV. TELNET ATTACK

		Suricata		Kitsune	
		Malicious	Benign	Malicious	Benign
Reality	Malicious	1.117	0	0	1.117
	Benign	0	113.384	2.848	110.536

D. Discussion

Our observations indicate that signature-detecting IDS find attacks more reliably than anomaly-detecting ones. However, the preconfigured set of signatures stems from well-researched attacks from the past. Novel attacks might pose a challenge for signature-detecting approaches, until the signatures are updated. Furthermore, a Smart Home installation might contain specific or rare Smart Home devices that are not considered in the preconfigured set of signatures. This is particularly problematic, as our role "‘private user"’ cannot be assumed to successfully define IDS signatures, but needs an expert. Our experiments have also shown that anomaly-detecting IDS might generate many false positives. This is because some of our Smart Home devices change their communication behavior from time to time. For example, the Echo Dot might switch from reading the weather report to playing music.

Furthermore, the learning phase of an anomaly-detecting IDS in a Smart Home is not monitored by an expert. If the Smart Home network is already compromised when the IDS is put into operation, this state is considered as normal. Thus, there exist situations in which anomaly-detecting IDS cannot increase the network security. We conclude that signature-based IDS are better-suited to secure Smart Homes at the moment. However, the pre-configuration needs special attention.

Note that IDS are able to detect more complex attacks than Portscans or Telnet attacks, even before such attacks were successful. Nevertheless, it is a challenge already to present the private user with an understandable solution for simple attacks, which can be implemented without expert knowledge. In the case of an unsuccessful attack, generic solutions such as "Please check with the manufacturer how to proceed in case of a security incident" cannot be applied. For this reason, we assume that for complex attack attempts, involving an expert via Reporting becomes even more important.

VI. Conclusion

To secure a Smart Home installation is challenging. Typically, private users do not possess the IT-Security expertise needed to implement adequate security measures. Furthermore, almost all Smart Home devices hide security-related details from the user and do not allow to inspect its software.

In this paper, we have developed a concept to implement an Intrusion Detection System into a Smart Home installation without violating the user's privacy, and without requiring the user to possess in-depth expertise. We have analyzed in which way the network segmentation, system architecture, IT-Security Process and the contractual liabilities of an IDS must be adapted for that purpose. We have tested our concept with a series of experiments on four different Smart Home devices. Our experiments have indicated that at this moment, signature-detecting IDS, such as Suricata are suitable to secure Smart Home installations. In contrast, anomaly-detecting IDS like Kitsune are problematic. The anomaly detection algorithms tend to misclassify changing user behavior as an attack, but the user lacks the expertise needed to rule out false alarms.

As a part of our future work we will consider specific situations which occur when new Smart Home devices are added to the network segment or the devices change their behavior after a functional update.

References

[1] TI Media Limited, "philips smart light," https://www.trustedreviews.com/best/best-smart-lighting-3600693, accessed: 2020-08-14.

[2] CBS Interactive Inc., "c-net smart gardening," https://www.cnet.com/news/smart-garden-buying-guide/, accessed: 2019-06-06.

[3] IBM, "Security intelligence," https://securityintelligence.com/news/latest-mirai-malware-variant-contains-18-exploits-focuses-on-embedded-iot-devices/, accessed: 2020-08-19.

[4] PR Newswire Association LLC, "New reaper iot botnet leaves 378 million iot devices potentially vulnerable to hacking," https://www.prnewswire.com/news-releases/new-reaper-iot-botnet-leaves-378-million-iot-devices-potentially-vulnerable-to-hacking-300542019.html, accessed: 2020-08-19.

[5] M. Gajewski, J. M. Batalla, G. Mastorakis, and C. X. Mavromoustakis, "A distributed ids architecture model for smart home systems," *Cluster Computing*, pp. 1–11, 2017.

[6] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home iot devices," *IEEE Internet of Things Journal*, pp. 9042–9053, 2019.

[7] B. L. R. Stojkoska and K. V. Trivodaliev, "A review of internet of things for smart home: Challenges and solutions," *Journal of Cleaner Production*, vol. 140, pp. 1454–1464, 2017.

[8] Wareable Ltd., "Amazon Echo voice control," https://www.the-ambient.com/guides/best-amazon-alexa-commands-280, accessed: 2019-02-25.

[9] C. Wilson, T. Hargreaves, and R. Hauxwell-Baldwin, "Benefits and risks of smart home technologies," *Energy Policy*, vol. 103, pp. 72–83, 2017.

[10] S. Kumar, "Survey of current network intrusion detection techniques," *Washington Univ. in St. Louis*, pp. 1–18, 2007.

[11] G. A. Marin, "Network security basics," *IEEE security & privacy*, vol. 3, no. 6, pp. 68–72, 2005.

[12] G. Carl, G. Kesidis, R. R. Brooks, and S. Rai, "Denial-of-service attack-detection techniques," *IEEE Internet computing*, vol. 10, no. 1, pp. 82–89, 2006.

[13] B. Harris and R. Hunt, "Tcp/ip security threats and attack methods," *Computer communications*, vol. 22, no. 10, pp. 885–897, 1999.

[14] H.-J. Liao, C.-H. R. Lin, Y.-C. Lin, and K.-Y. Tung, "Intrusion detection system: A comprehensive review," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 16–24, 2013.

[15] A. Lazarevic, K. Vipin, and S. Jaideep, "Intrusion detection: A survey," in *Managing Cyber Threats. Massive Computing*. Springer, 2005, pp. 19–78.

[16] A. K. Saxena, S. Sinha, and P. Shukla, "General study of intrusion detection system and survey of agent based intrusion detection system," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*. IEEE, 2017.

[17] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *computers & security*, 2009.

[18] G. Disterer, "Iso/iec 27000, 27001 and 27002 for information security management," 2013.

[19] Federal Office for Information Security, "BSI-Standard 200-2, IT-Grundschutz-Methodology," *https://www.bsi.bund.de*, 2017.

[20] O. of Government Commerce, *Introduction to ITIL*. Van Haren Publishing, 2005.

[21] J. Eloff and M. Eloff, "Information security architecture," *Computer Fraud & Security*, 2005.

[22] Federal Office for Information Security, "BSI-Standard 200-1, Information Security Management Systems (ISMS)," *https://www.bsi.bund.de*, 2018.

[23] S. S. I. Samuel, "A review of connectivity challenges in iot-smart home," in *2016 3rd MEC International conference on big data and smart city (ICBDSC)*. IEEE, 2016, pp. 1–4.

[24] R. A. Kemmerer and G. Vigna, "Intrusion detection: a brief history and overview," *Computer*, vol. 35, no. 4, pp. supl27–supl30, 2002.

[25] W. Ali, G. Dustgeer, M. Awais, and M. A. Shah, "Iot based smart home: Security challenges, security requirements and solutions," in *2017 23rd International Conference on Automation and Computing (ICAC)*. IEEE, 2017, pp. 1–6.

[26] N. Gupta, V. Naik, and S. Sengupta, "A firewall for internet of things," in *2017 9th International Conference on Communication Systems and Networks*. IEEE, 2017, pp. 411–412.

[27] B. Zhang, P.-L. P. Rau, and G. Salvendy, "Design and evaluation of smart home user interface: effects of age, tasks and intelligence level," *Behaviour & Information Technology*, vol. 28, no. 3, pp. 239–249, 2009.

[28] C. Beckel, H. Serfas, E. Zeeb, G. Moritz, F. Golatowski, and D. Timmermann, "Requirements for smart home applications and realization with ws4d-pipesbox." IEEE, 2011, pp. 1–8.

[29] S. Zamfir, T. Balan, I. Iliescu, and F. Sandu, "A security analysis on standard iot protocols," in *2016 International Conference on Applied and Theoretical Electricity (ICATE)*. IEEE, 2016.

[30] SolarWinds Worldwide, "7 Best Intrusion Detection Software and Latest IDS Systems," https://www.dnsstuff.com/network-intrusion-detection-software, accessed: 2020-06-18.

[31] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: an ensemble of autoencoders for online network intrusion detection," *arXiv preprint arXiv:1802.09089*, 2018.

[32] wolfSSL Inc., "wolfMQTT Client Library," https://www.wolfssl.com/products/wolfmqtt, accessed: 2020-8-14.