Introducing the Cyber-Physical Data Flow Diagram to Improve Threat Modelling of Internet of Things Devices

Simon Liebl [1,2], Ian Ferguson [2], Andreas Aßmuth [3], Natalie Coull [2], George R. S. Weir [4]
[1] emgarde, Ebermannsdorf, Germany
e-mail: simon.liebl@emgarde.de
[2] Abertay University, Dundee, UK
e-mail: {i.ferguson | n.coull}@abertay.ac.uk
[3] Kiel University of Applied Sciences, Kiel, Germany
e-mail: andreas.assmuth@haw-kiel.de
[4] University of Strathclyde, Glasgow, UK
e-mail: george.weir@strath.ac.uk

***Abstract*—A growing number of Internet of Things (IoT) devices are used across consumer, medical, and industrial domains. They interact with their environment through sensors and actuators and connect to networks such as the Internet. Because sensors may collect sensitive data and actuators can trigger physical actions, security, privacy, and safety are major challenges. Threat modelling can help identify risks, but established IT-focused methods transfer to the IoT only to a limited extent. In this paper, a new modelling technique specifically for IoT devices called Cyber-Physical Data Flow Diagram (CPDFD) is proposed that also allows modelling of hardware with the aim to support manufacturers in identifying threats and developing countermeasures. The technique was examined through an experimental study and a survey with interviews. The results suggest that numerous other attack scenarios can be found through the modelling technique, improving the identification of threats to IoT devices.**

Keywords-Security; Internet of Things; Embedded Systems; Threat Modelling; Cyber-Physical Data Flow Diagram.

I. Introduction

The Internet of Things (IoT) is the interconnection of billions of devices via the Internet or another network. These devices are used in a wide variety of areas, such as the smart home, medical, and industrial applications. Their key characteristics include interacting with the environment via sensors and actuators and communicating with other devices and systems. Networking them aims to extend functionality, integrate services, improve usability, increase efficiency, and reduce costs. However, adding multiple network interfaces also expands the attack surface and increases the risk of cyber attacks. It seems that the number of published incidents is steadily increasing [1], indicating that IoT security is a serious problem. In recent years, incidents have been reported in various IoT application fields, such as industrial (e.g., attacks on Ukraine's power grid [2]), medical (e.g., insulin pumps and ventilators [3]), automotive (e.g., hack of Tesla and its Wall Connector [4][5]), infrastructure (e.g., traffic lights [6]), and smart home (e.g., vacuum robot [7]).

The described small sample of security incidents related to IoT devices and systems shows that there is a systematic problem threatening security, privacy, and safety. Therefore, new approaches are required to better protect IoT devices from such threats. A promising approach that is already established in the IT domain is the risk assessment using threat modelling. Threat modelling is the systematic approach to identifying threats and vulnerabilities to a particular system. Appropriate countermeasures can then be defined. The process can also be carried out early in the product lifecycle, which supports the goal security by design. However, common threat modelling approaches from the IT domain cannot necessarily be adopted to IoT devices. Reasons for this are that IoT devices consist of resource-constrained embedded systems and that their interaction with the real world poses an increased risk to the privacy and safety of users. As part of this work, the modelling technique Data Flow Diagram (DFD), which is often used in threat modelling, was specifically examined. Several adaptations are proposed to increase the applicability of DFDs to IoT devices, including two new elements and the possibility to create a Hardware Diagram (HWD). This extension, called Cyber-Physical Data Flow Diagram (CPDFD), has the aim to enable more detailed modelling of the components of IoT devices and thus allow better threat identification compared to regular DFDs. This technique is intended to support device manufacturers in identifying threats and developing appropriate countermeasures.

The remainder of this paper is structured as follows: in Section II, the fundamentals of IoT devices and threat modelling are provided. Section III presents related work regarding IoT security, threat modelling, and DFDs. In Section IV, the proposed extension CPDFD is introduced. The methodology for evaluating CPDFDs consisting of two studies is described in Section V, followed by the presentation of the results. These results are then discussed in Section VII. The paper ends with conclusions in Section VIII.

II. Background

A. Internet of Things Devices

The IoT is a "group of infrastructures interconnecting connecting objects and allowing their management, data mining and the access to the data they generate" [8]. These objects are devices extended with network connectivity and computing capabilities and require usually minimal human intervention [9]. The architecture of the IoT is often structured into different

 ISBN: 978-1-68558-442-9

layers [10]. The perception layer, the lowest layer, consists of these physical objects, such as sensors, actuators, or Radio-Frequency Identification (RFID) tags. These device are also used in a Cyber-Physical System (CPS), for example, to control physical processes in the real world [11]. For such systems, real-time requirements are essential, which differentiates them clearly from IT systems. The impact of IoT devices can thus affect security, privacy, and safety and have consequences such as the disclosure of sensitive data in medical applications or human injury in industrial applications.

IoT devices are embedded systems, which are specific computer systems built for a custom purpose. Their basic components consist of hardware, software, and data [12]. Examples for hardware components are microcontroller, memory chip, Printed Circuit Board (PCB), security chip, power supply, and various sensors and actuators. Software components include firmware, basic libraries for cryptography and logging, and protocol stacks for numerous IoT protocols, such as Bluetooth, ZigBee, and MQTT. Besides firmware, devices store access data, keys, and configuration and log files, among others.

B. Threat Modelling

In this section, the DFD and two published threat modelling techniques are introduced. The small excerpt of techniques is necessary to understand the remainder of this paper.

1) Data Flow Diagram: The DFD is a graphical modelling technique used historically in the field of software engineering and system analysis [13]. It quickly gained popularity due to its intuitive nature and ability to capture both high-level and detailed views of system processes and data. A DFD consists only of four elements: *Process*, *Data Store*, *External Entity*, and *Data Flow*. This enables a simple visual representation of complex systems and illustrates how data is input, processed, stored, and output. Processes represent the transformation of data and can be thought of any running code. Data stores depict any type of data storage, such as files or databases. External entities are used to describe external data sources or destinations, e.g., people or any code outside your control. Data flows are used to connect the other three elements. Besides their use in software engineering, security experts found their application also valuable in the field of cyber security. The analysis of the system is an important part of a security assessment conducted to find threats and vulnerabilities. In order to limit the assessment for complex systems, another element called *Trust Boundary* or *Trust Area* was added to visualise different trust levels, initiating the second version of DFDs [14]. Since DFDs are typically created during early design, they naturally support the analysis of architectural security issues rather than implementation-level defects.

2) STRIDE and LINDDUN: The CIA triad – Confidentiality, Integrity, and Availability – forms the basis of information security, often complemented by goals such as authenticity, non-repudiation, and authorisation. STRIDE covers the threats spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege [14], and was developed at Microsoft to identify such threats during design [15]. It maps directly to these six security goals.

STRIDE can be applied to DFDs, but checking all six threats for every element is time-consuming. Since some elements are more susceptible to specific threats, STRIDE-per-Element and STRIDE-per-Interaction were introduced to streamline the analysis [14].

LINDDUN is a framework for privacy threat modelling. Similar to STRIDE, LINDDUN is an acronym that stands for linkability, identifiability, non-repudiation, detectability, disclosure of information, unawareness, and non-compliance and can be combined with DFDs as well [16]. The threats are therefore not focused on security goals, but are instead aimed at privacy goals.

III. Related Work

IoT security and privacy is still a huge problem, which is why a lot of research is being conducted in a variety of directions. Due to the many articles around IoT security, there are several reviews with the aim to summarise, among others, threats, attacks, challenges, and solutions [17]–[21]. Some researchers focused specifically on IoT devices, often using hands-on approaches to show how embedded systems can be attacked, which threats arise, and how they can be mitigated [22]–[26]. Regarding threat modelling and risk assessment, there are a couple of articles with the aim of adapting these methods for the IoT. In [27], they aim to automate this process for IoT systems by providing their own methodology. The authors of [28] present a STRIDE and DFD-based threat modelling approach specifically for CPSs with special focus on human injury, equipment damage as well as black-out. Many articles address specific fields of application such as automotive [29], building and home automation [30], agriculture [31], and healthcare devices [32]. DFDs are also relevant to this work. In this regard, [33] and [34] are particularly worth mentioning, as they propose adjustments to DFDs with the aim of improving threat identification and generation.

In summary, there is a lot of research in the area of IoT security and privacy, also with special emphasis on threat modelling. In the case of IoT devices in particular, the differences between embedded systems and IT systems were recognised and threats analysed that emerge through sensors, actuators, and interaction with the environment and other systems. However, the hands-on approaches and the various methods for the different IoT applications emphasise that a common technique for modelling IoT devices has not yet been established.

IV. Cyber-Physical Data Flow Diagram

This section introduces the proposed modelling technique CPDFD by describing its aims and adaptations.

A. Aim

As mentioned before, there are several issues with modelling IoT devices. One big difference between IT and IoT applications is the interaction with the physical world. While this is an

integral part of the latter, it is hardly present in IT applications. In IoT systems, a sensor, the data source, usually measures an environmental value, such as temperature, photographic picture, or heart activity. The processing of the data can in turn lead to physical operations by the actuator, the data sink, and thus result in changes in the environment. For example, a valve is opened, a car battery is charged, or a door is unlocked. From a security point of view, these sensor values and actuator actions are particularly interesting, as they can threaten the privacy and safety of users. It is therefore necessary to address this difference, the interaction with the physical environment, also in the modelling.

Another difference between IT and IoT applications is the location and purpose. IoT devices are used in critical applications, such as healthcare, automotive, and automation. Many of them are placed outdoors, e.g., security cameras, charging stations, or pipeline valves. The criticality as well as the accessibility of the devices make hardware attacks interesting for attackers and are frequently reported [35]. Successful attacks give deep insight into the device and often lead to access to Intellectual Property (IP), sensitive data, credentials, and cryptographic secrets. Therefore, hardware attacks, which are often out of scope in IT applications, need to be considered due to the accessibility and the frequent critical application.

One core element of IoT devices is the communication with other devices and systems. Besides dozens of wireless communication protocols, such as Z-Wave, Wi-Fi, and NFC, wired protocols including Modbus, Ethernet, and USB are commonly used. Additionally, several common protocols used in embedded systems are used, for instance, UART, RS-232, and JTAG. With the multitude of protocols and interfaces in use, it is therefore easy to lose track which of them are actually utilised in an IoT device. However, these various protocols and interfaces increase the attack surface and can become a gateway for attackers. The goal is therefore to be able to represent these interfaces in a model in order to get a quick overview of the connectivity of a device.

There is another difference between the development of IoT devices and IT applications. Modelling the latter is mainly done by software and network engineers that commonly have a computer science background. However, modelling an IoT device may also require the help of hardware and embedded engineers with a background frequently in mechanical or electrical engineering. Therefore, a modelling technique for IoT devices needs to be simple and at the same time comprehensive in order to be applicable by people with different backgrounds.

B. The Technique

As mentioned before, the CPDFD is proposed to support modelling of IoT devices. The technique extends the DFD, which is commonly used for security assessments and is known by many security experts and engineers. The following paragraphs describe adjustments to the DFD in order to achieve the previously described aims.

It is difficult to represent the interaction with the physical environment through sensors and actuators using the five elements of a DFD. Depending on the sensor characteristics (e.g., simple electrical resistors or modules with a serial interface), it can make sense to represent them as *Process*, *Data Store*, or *External Entity* or even as a combination with *Data Flows*. However, this question of detail would take a lot of time and contradict the principle of simplicity of DFDs. If one does not know how to represent a specific component right away, it could be omitted. This in turn would lead to a situation where measured values which might threaten the privacy, for example, would not be recorded. This applies analogously to actuators and safety as a protection goal. It is therefore proposed to add a new element called *Physical Link* to the DFD notation in order to be able to model these interactions with the physical environment. The goal of this element is to give sensors and actuators a first-class representation in the diagrams, highlighting issues with the goals of privacy and safety.

Another aim of the CPDFD technique is to enable the creation of hardware models in order to support the identification of hardware attacks, among others. In theory, there are dozens of modelling techniques (see [36]). However, the authors are not aware of any hardware modelling technique that is commonly used for the threat analysis. Hardware components were integrated into DFDs in a few cases (e.g., [37][38]). It is proposed to use the DFD notation to create a separate HWD. This is achieved by splitting the elements into logical and physical ones. Regular DFDs are mainly used for modelling software-related components, which are now referred to as logical elements. The additional physical elements are used for the creation of the HWD. The physical data store can be used to model a flash or One Time Programmable (OTP) memory, for instance. Likewise, the physical processor and the physical trust area can be used to model microprocessors and PCBs, for example. Modelling of data flows is not necessary in the HWD.

The CPDFD technique should also allow users to get a quick overview of the device's communication interfaces. Similar to the *Physical Link* before, modelling the interfaces through one of the standard elements could have the effect that these are not modelled. It is therefore proposed to model them as a separate element named *Interface* as part of the HWD.

All elements of a CPDFD and their use in a DFD and HWD are shown in Table I. The proposed extension has further advantages. A DFD can be augmented with links to the hardware model. For example, for data stores, such as a file, it can be specified in which memory it is stored and for data flows, the utilised communication interface can be provided. This additional context can lead to a more precise description of an attack scenarios and thus support the threat identification. The new elements *Physical Link* and *Interface* have another benefit when it comes to software-aided modelling. Threat rules for automatic threat generation can be specified more precisely or can be created specifically for the two elements. This can reduce false positives, i.e., wrongly identified threats,

TABLE I. THE ELEMENTS OF A CPDFD. THE LAST TWO COLUMNS INDICATE THEIR USE IN DFDS AND HWDS.

Element	Symbol	Meaning & Examples	DFD	HWD
Process	Name	Executed code Log. Process: Code in C, web server Phy. Processor: Microprocessor, TPM	☒ ☐	☐ ☒
Data Store	Name	Things that store data Log. Data Store: File, database Phy. Data Store: FLASH, OTP memory	☒ ☐	☐ ☒
External Entity Interactor	Name	People or code outside your control Log. External Entity: Browser, cloud service Phy. External Entity: User, machine	☒ ☒	☐ ☐
Data Flow	Name Name	Communication between elements (Log.) Data Flow: Data flow with protocol (stack)	☒	☐
Trust Area	Name	Delimited area of trust Log. Trust Area: Net segment, container Phy. Trust Area: PCB, casing	☒ ☒	☐ ☒
Physical Link	Name	Link between physical and logical world Physical Link: Microphone, battery, motor	☒	☒
Interface	Name	Communication interface Interface: Wi-Fi, USB, JTAG	☐	☒

and false negatives, i.e., undetected threats. The commonly used techniques STRIDE-per-Element and LINDDUN can still be used for CPDFDs and also provide more accurate results. Last but not least, the CPDFD technique establishes a unifying notation for HWDs and DFDs. The resulting diagrams remain simple and clear, despite the addition of two elements and the distinction between physical and logical elements. They can be created and understood by hardware, embedded, and software engineers and other people as well.

V. METHODOLOGY

The methodology for evaluating the improvements of the CPDFD is introduced in this section. It consists of an experimental study and a survey with interviews.

A. Experimental Study

This section introduces an experimental study with the objective to compare CPDFDs with DFDs by quantitatively comparing the number of attack scenarios identified by two groups, each using one modelling technique. Participants of the study, who were students of computer science-related study programmes, were put in the following situation: As innovative students, they have applied their previously acquired knowledge in practice and developed a smart IoT device alongside their studies. They now want to bring the device to market, but they still have concerns about security and privacy. Therefore, each participant had the task of examining the device for threats and vulnerabilities. The analysed device was the open source device Jaimico [39], which is a combination of voice assistant and health monitor. The companion robot can be worn on the shoulders in daily life and can be used as a voice assistant or health monitor to measure body temperature and heart activity by connecting to a wearable. In addition to analysing health data in the cloud, the device offers the special feature of detecting Covid19 through analysing recorded coughing and body temperature. In this study, participants were supported by a custom software tool called TTModeler [40][41], but the supported modelling technique, specifically the two new elements, differed for both groups. The modelling technique thus serves as an independent variable. The produced project file and a concluding questionnaire served as the data source for further analysis. For the sake of completeness, it should be noted that there was a third group with a different software tool, but this is not of interest for this paper.

The aim of this study is to gather evidence on the usability of each tool, which is the "degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [42]. For this purpose, effectiveness is defined as "accuracy and completeness with which users achieve specified goals" and efficiency as the resources expended in relation to the effectiveness [42]. More precisely, it was measured how much time is spent on the creation of a specific model and how many attack scenarios were found in this context, enabling the quantification in identified attack scenarios per minute for the hardware model, among others. This can be used to determine the efficiency. The impact of the additional elements *Physical Link* and *Interface* was specifically examined.

It was decided to conduct the study with students because they are available at short notice and in larger numbers than, for example, developers from industry. Students from study programmes related to computer science were invited to participate in the study. All participating students were in the fourth year of their studies. They represent the actual users well, as they have, for example, only rudimentary experience in the field of security, as it is the case with many firmware and software developers. A total of 46 students volunteered to participate in the study. Five of them took part in a preliminary study, which was conducted to find issues in the procedure and task description, and their results are not considered.

This results in a sample size of 41. Due to the omitted third group and unexpected cancellations, 12 participants used the modelling technique CPDFD, denoted as G_{CPDFD}, and 14 participants used the DFD technique (G_{DFD}).

Participants first read an information document covering the CIA, STRIDE, and LINDDUN threat models, as well as IoT privacy threats defined by [43]. They then received a second document introducing DFDs, including STRIDE-per-Element, LINDDUN-per-Element, and example diagrams. Group G_{CPDFD} received a modified version explaining CPDFD elements instead of standard DFDs. Next, participants read the description of the example device Jaimico and watched a video demonstrating the main features of the assigned tool, including how to model the "Register device" use case and add threats. They then worked on the task, with the option to stop at any time within a three-hour limit. Finally, they completed the questionnaire and submitted their project file.

The submitted project files and questionnaires served as the data sources. Project files were analysed for diagrams, elements, and attack scenarios. Duplicates and out-of-scope scenarios were marked and excluded. Remaining scenarios were filtered and labelled as considered attack scenarios. Because the device description was partly fictitious, scenarios were classified as having either low or high probability of correctness. Most were rated high, with exceptions such as "Malicious file on USB flash drive", since the USB port was specified as charging-only. Each scenario was assigned a type: *generic*, *outlined*, or *custom*, indicating the level of detail and difficulty of identification. *Generic* scenarios were directly derived from STRIDE or LINDDUN, *outlined* scenarios added some detail (e.g., "Clear text data storage of the database"), and *custom* scenarios were highly specific or not obviously derived from a threat model (e.g., "Unsecure Bluetooth version"). Only high-probability scenarios of type *outlined* or *custom* were considered relevant.

The tool was modified to track modelling time. After completing the task, participants answered a 21-item questionnaire, including eight items on threat identification and 13 on usability.

The further analysis of the collected data was carried out with the help of descriptive and inferential statistical methods [44] and may be visualised using a box plot [45]. The significance level for rejecting a null hypothesis is set at $\alpha = 5\%$. Depending on whether the results are normally distributed, either the *t*-test or the Mann-Whitney *U* test is used to test for a difference between two groups. Subsequently, Cohen's d is calculated to quantify the effect size [46][47]. It can be assumed that there is a small effect for $0.2 < d < 0.5$, a medium effect for $0.5 \leqslant d < 0.8$, and a large effect for $d \geqslant 0.8$.

B. Survey and Interviews

The last study aims to get feedback from people who might actually utilise CPDFDs. These include device manufacturers as well as consultancies. Manufacturers often do not have security experts themselves and therefore hire external companies. Participants apply the methodology (Thing Threat Modelling (TTM)), technique (CPDFD), and tool (TTModeler) on their own devices and provide afterwards feedback through a questionnaire and an interview. The aim is to specifically ask for opinions on the HWD and the new elements.

TABLE II. OVERVIEW OF PARTICIPANT AND COMPANY CHARACTERISTICS.

Category	Distribution
Company type	8 device manufacturers; 7 service providers
Participant role	9 security experts; 6 engineers
Company size	7 large; 3 medium-sized; 4 small; 1 micro
Field of application	8 industrial; 3 consumer; 2 automotive; 1 medical; 1 infrastructure
Experience	46.7% embedded security; 80.0% DFD; 60.0% threat modelling

All persons with a connection to cyber security, IoT, or embedded systems were invited for the study. In total, 15 participants could be recruited for the study. Table II categorizes participants according to the type of company, the size of the company (according to [48]), and the experience. Noteworthy is that two participants had no cyber security experience (13.3%).

At the start of the study, participants received an introduction to threat modelling, the adaptations used in this work, and their task. After completing and evaluating the contributions, they filled out a questionnaire and answered a few interview questions. The questionnaire, consisting mainly of five-point Likert items [49] and free-text fields, together with the interview, served as the data sources. Questionnaire responses were analysed descriptively, while the free-text fields and interview data were examined qualitatively.

VI. RESULTS

The following section summarises the results of both studies.

A. Experimental Study

This section presents the results of the experimental study, summarised in Table III. The first tests on the results using the Shapiro-Wilk test showed that the sample is not normally distributed. This could be caused by the rather small sample size, for example. It was therefore decided to consistently use non-parametric methods.

Table III shows the duration needed to conduct the study for both groups. Note that this duration represents the pure interaction time with the tool and does not include the learning time required for the threat models, CPDFD/DFD, and the introduction to Jaimico. Group G_{CPDFD} has a wide range of 80 to 142 minutes ($M = 111$ min, $SD = 20$ min). In contrast, G_{DFD} has a higher mean value of 120 minutes ($SD = 10$ min) and a narrower range of 98 to 132 minutes. Despite these differences, there is a large overlap of the ranges and it indicates that the technique used does not have a significant influence on the execution duration (Mann-Whitney *U* test, $U = 62.5$, $p = .280$, $p > \alpha$).

The number of considered attack scenarios, i.e., without duplicates, high probability of correctness and out of scope (see subsection V-A), have a large range. Participants of G_{CPDFD} found in total between 2 and 182 attack scenarios ($M = 69.21$, $SD = 40.44$) and G_{DFD} found between 2 and 74 scenarios ($M = 45.17$, $SD = 19.85$). The scenarios start at 2, as some

TABLE III. OVERVIEW OF THE PROJECT FILE ANALYSIS.

	G_{CPDFD}	G_{DFD}
Duration	111 min ± 20 min	120 min ± 10 min
Considered attack scenarios	69.21 ± 40.44	45.17 ± 19.85
Type: Generic	19.29 ± 22.08	17.92 ± 14.64
	27.86% ± 31.90%	39.67% ± 32.41%
Type: Outlined	10.86 ± 5.84	9.58 ± 7.95
	15.69% ± 8.44%	21.22% ± 17.60%
Type: Custom	39.07 ± 17.17	17.67 ± 10.00
	56.45% ± 24.80%	39.11% ± 22.15%
Relevant attack scenarios	49.93 ± 21.00	27.25 ± 11.32
	72.14% ± 30.34%	60.33% ± 25.07%
Goal: Security	44.93 ± 19.48	27.00 ± 11.14
	89.99% ± 39.01%	99.08% ± 40.89%
Goal: Privacy	5.00 ± 2.11	0.25 ± 0.45
	10.01% ± 4.23%	0.92% ± 1.66%
Physical Link-related	10.86 ± 5.17	
	21.75% ± 10.36%	
Interface-related	11.21 ± 5.10	
	22.46% ± 10.22%	
Model: Hardware	29.07 ± 11.28	9.50 ± 6.67
	58.23% ± 22.58%	34.86% ± 24.47%
Model: Software	4.29 ± 8.09	4.67 ± 5.77
	8.58% ± 16.20%	17.13% ± 21.19%
Model: Data flow	16.57 ± 8.71	13.08 ± 9.40
	33.19% ± 17.44%	48.01% ± 34.51%
Time for hardware model	26 min ± 14 min	12 min ± 12 min
Time for software model	9 min ± 7 min	8 min ± 7 min
Time for data flow model	64 min ± 21 min	96 min ± 17 min
Time per attack scenario	2 min ± 0 min	4 min ± 0 min
- for hardware model	1 min ± 0 min	1 min ± 1 min
- for software model	2 min ± 2 min	2 min ± 2 min
- for data flow model	4 min ± 1 min	7 min ± 1 min

Note: The values of the last two columns show the mean value and the standard deviation. Furthermore, the table is split in three sections. The percentages in the second section refer to the considered attack scenarios, those in the third section to the relevant attack scenarios.

participants had mainly modelled the example use case showed in the introductory video. Their type, *generic*, *outlined*, and *custom*, can be further analysed. The results for the types *generic* and *outlined* are similar and with overlapping error bars (see Table III). A difference can be seen for the type *custom*. The groups G_{CPDFD} and G_{DFD} found on average 39.07 (SD = 17.17) and 17.67 (SD = 10.00) scenarios respectively. The Mann-Whitney U test shows that both groups are significantly different from each other. Therefore, G_{CPDFD} found significantly more attack scenarios of the high ranked type *custom* than G_{DFD} (U = 155.0, p = .000, $p \leqslant \alpha$, d_{Cohen} = 1.50, large effect). This means that G_{CPDFD} found more scenarios with high information value that are more difficult to find.

The attack scenarios were filtered again to include only those of the type *outlined* and *custom* as explained in subsection V-A. This reduced list of relevant attack scenarios is visualised in Figure 1. The main range of G_{CPDFD} is between 33 and 74 scenarios with two outliers at 2 and 94, while G_{DFD} ranges between 2 and 39. The ranges overlap, but the boxes highlighting the lower and upper quartiles do not. The median of G_{CPDFD} (49) differs from G_{DFD} (30) by 19 scenarios. The result of the Mann-Whitney U test shows that group G_{CPDFD} identified significantly more attack scenarios than G_{DFD} (U = 147.0, p = .001, $p \leqslant \alpha$, d_{Cohen} = 1.32, large effect). This clearly shows that group G_{CPDFD}, which had the *Physical Link*

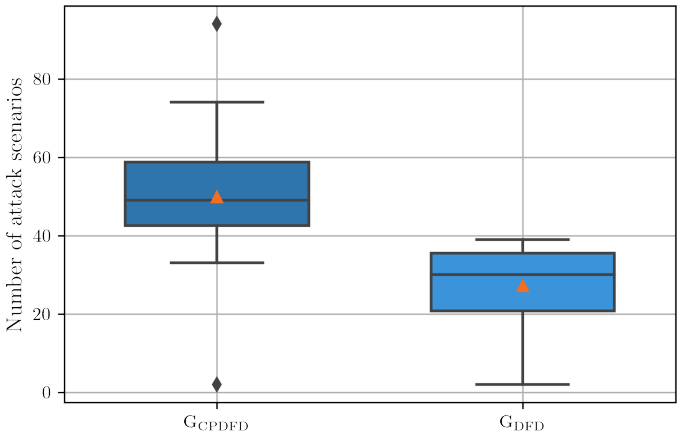

Figure 1. Relevant attack scenarios for groups G_{CPDFD} and G_{DFD} as box plot.

and *Interface* available, identified substantially more attack scenarios.

Attack scenarios were categorised whether these threaten the security or privacy of the device under consideration. Group G_{CPDFD} identified 5.00 scenarios on average (SD = 2.11), while G_{DFD} identified less than 1 scenario on average (M = 0.25, SD = 0.45). Therefore, they identified significantly more privacy attack scenarios than G_{DFD} (Mann-Whitney U test, U = 160.5, p = .000, $p \leqslant \alpha$, d_{Cohen} = 3.01, large effect). A more detailed analysis shows that out of these 5 scenarios of G_{CPDFD}, 3.64 scenarios were identified through the element *Physical Link* (SD = 1.69), 1.21 through the *Interface* (SD = 0.58), and only 0.14 scenarios through other elements (SD = 0.36). Consequently, this suggests that the used technique, CPDFD or DFD, has an influence on the identified attack scenarios against privacy.

Figure 2 shows the average number of identified attack scenarios per group. Furthermore, it highlights how many scenarios of G_{CPDFD} relate to the additional elements of a CPDFD – *Physical Link* and *Interface*. The comparison of G_{CPDFD} and G_{DFD} shows that both groups found on average almost the same number of scenarios for category *Others*. The extra scenarios of G_{CPDFD} relate to the new elements. Therefore, this suggests that further attack scenarios can be found through the two elements.

Table III includes in which model, hardware, software, or data flow, the attack scenarios were identified. The main difference can be seen for the hardware model. Group G_{CPDFD} has a range of 14 to 45 identified scenarios, without a single outlier at 2, and a median of 29.5. G_{DFD} has a range of 0 to 19 and a median of 11.5. The Mann-Whitney U test shows that the groups are significantly different (U = 156.0, p = .000, $p \leqslant \alpha$, d_{Cohen} = 2.08, large effect). Again, the difference between G_{CPDFD} and G_{DFD} arises from the two new elements. The median *Physical Link* and *Interface*-related scenarios for the former are 6 and 10.5 scenarios, respectively. The median scenarios for the standard elements are 13 and similar to G_{DFD}. The results indicate that it is worth creating a hardware model and that there is an improvement due to the new elements.

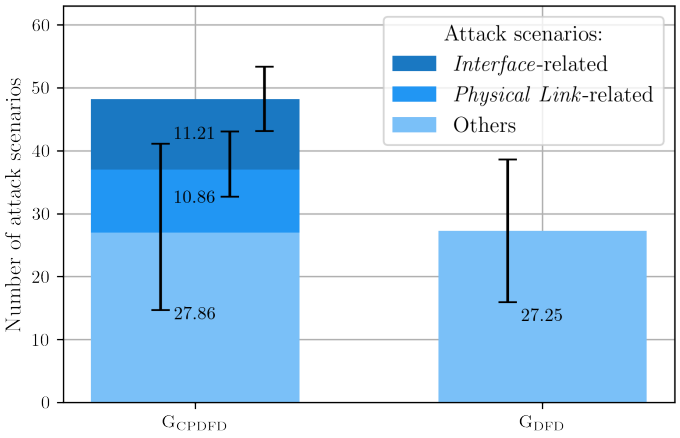

Figure 2. Average attack scenarios per group. In group G_{CPDFD}, scenarios relating to the elements *Physical Link* and *Interface* are highlighted.

Furthermore, the tool TTModeler tracked how much time participants spent on creating and analysing the different models, enabling the calculation of the duration per attack scenario (see Table III). Group G_{CPDFD} needed on average 2 minutes per scenario ($SD = 0$ min), while G_{DFD} needed 4 minutes ($SD = 0$ min). The average time per attack scenario in the hardware model was even less for both groups, because both needed 1 minute per scenario on average. These scores indicate the improved efficiency of CPDFDs and also of a hardware model in general.

It was recorded how participants modelled certain components. The technique CPDFD would classify microphone, loudspeaker, touch screen, and temperature sensor as element *Physical Link*, for example. While almost all participants in group G_{CPDFD} modelled the microphone as *Physical Link* (93%), the other group was divided. One quarter modelled the microphone as *Process*, 17% as *External Entity*, and the majority of 58% did not model it at all. For the loudspeaker, the picture is almost the same: only one participant had modelled it as *Data Store* instead of *Process*. While the majority of G_{CPDFD} also modelled the touch screen, the temperature sensor, and the PCB, the share for G_{DFD} was one third or less. The battery was not modelled by anyone of G_{DFD}, but by 86% of G_{CPDFD}. Likewise, the same was analysed for the interfaces JTAG, Bluetooth, Wi-Fi, and USB. Almost all participants of G_{CPDFD} modelled these components as element *Interface* (86%, 93%, 86%, and 79%, respectively). In contrast, only 8% of G_{DFD} modelled JTAG, Bluetooth, and Wi-Fi and 25% modelled the USB interface. Since the majority of G_{CPDFD} modelled all these components, while G_{DFD} did not, this suggests an added value of the two elements *Physical Link* and *Interface*.

Last, a brief review of the questionnaire. The only item of interest in this study was about the simplicity of creating a CPDFD/DFD. A total of 61.54% of G_{CPDFD} agreed that creating a DFD is easy, while 41.67% agreed of G_{DFD}. This seems to indicate that the two new elements of a CPDFD have not increased the difficulty of a DFD, but actually made it easier. However, this difference is not significant (Mann-Whitney U test, $U = 13.5$, $p = .911$, $p > \alpha$). Therefore, the difficulty of creating a CPDFD and DFD can be considered as equal.

B. Survey and Interviews

This section presents the results of the survey and interviews with device manufacturers and consulting companies. The precise duration for conducting the study was not measured. Participants indicated that they invested from two hours up to two days. Most of them applied the TTM methodology on their own devices. A few also carried out the analysis for other devices, e.g., power inverter or smart home. Some people used the results from previous assessments for a side-by-side comparison.

a) Questionnaire: All participants agreed that it is worth to create and analyse a hardware model (13 strongly agreed, 2 agreed) and no one disagreed that reusing the elements of a DFD is the right technique for this (4 strongly agreed, 5 agreed, 3 indecisive). It was also asked about the added value of the two new elements, resulting in exactly the same outcome. The majority of 58.3% strongly agreed that both elements *Physical Link* and *Interface* provided added value (17% agreed, 8% each indecisive, disagreed, and strongly disagreed). They also agreed (72.7%) that the elements contribute to a more accurate modelling of IoT devices. Additional feedback was provided noting that the two elements facilitate modelling, but that the interface needs to be more integrated. It was also remarked that the meaning of both elements is similar. The results suggest that the hardware model, the *Physical Link*, and the *Interface* improve the threat identification.

b) Interviews: This section summarises the findings of the interview analysis. Statements by individual persons are anonymised. If necessary, the person is described by categories introduced in subsection V-B. In these cases, the role of the persons and their company type are indicated in parentheses, together with their participant number, e.g., (P_1, security expert, manufacturer).

When participants were asked about the relevance of a hardware model, all agreed that this is an important part. Several participants stated that the hardware model was the best part of the TTM methodology. Hardware attacks are not considered sufficiently and "without them one cannot find the crux of the matter" (P_8, security expert, manufacturer). One participant had tried to represent hardware security in a model in the past using the system modelling language. However, he said that a structured threat analysis was not possible. In general, it was seen as positive that one can quickly see how the device is constructed and which interfaces and components, such as Trusted Platform Modules (TPMs), it uses.

The extension of the DFD notation with the two elements *Physical Link* and *Interface* was seen positive, but some participants were not fully convinced. One participant found the extension very good, as she "struggled in the past to map certain hardware components in a well-distinguishable yet abstract way" (P_{13}, security expert, service provider). The *Physical Link* was found to be good, as data is generated there and

actuators can influence the environment. “Even if a sensor is analogue, that does not mean you don‘t have to think about it” (P_{10}, security expert, service provider). More feedback was given for the *Interface*. It is useful and important, as “debug interfaces are often overlooked” (P_9, security expert, service provider). In a certain way, one is also forced to think about interfaces. However, a few participants noted that they did not fully understand the difference between both elements. The elements could also make the models more abstract and thus more vague in certain cases. One participant noted that the new elements are useful; their use is left to the user.

The feedback from the participants clearly shows that the hardware model brings improvements and is important for the threat analysis. Overall, the two new elements were also seen positively and their enrichment appreciated.

VII. Discussion

A. Hardware Model

In the experimental study, the median of group G_{CPDFD} identified 29.5 attack scenarios in the hardware model (59.1% of all), G_{DFD} found at least 11.5 scenarios (42.2%). The efficiency was also higher for the hardware model (1 minute per scenario) than for the average (2 and 4 minutes per scenario for G_{CPDFD} and G_{DFD} respectively). In the survey, all participants, security experts and engineers, agreed that it is worth to create and analyse a hardware model (see subsection VI-B). Furthermore, they reaffirmed the importance of the hardware model in the interviews. One of the advantages they mentioned was the quick overview of all components and interfaces. In conclusion, the quantitative and qualitative results speak in favour of a hardware model. This suggests that using a hardware model may improve threat identification.

B. Physical Link and Interface

The two new elements *Physical Link* and *Interface* were evaluated separately, but the jointly collected results differ only slightly. Their results are therefore summarised together first and then briefly discussed individually.

In the experimental study, the two groups G_{CPDFD} and G_{DFD} had the single difference that the former had the two new elements available in the diagram editor. However, G_{CPDFD} found significantly more scenarios than G_{DFD} (49.9 and 27.3 on average respectively) (see subsection VI-A). The assignment of the scenarios to elements showed that G_{CPDFD} also discovered 27.9 scenarios for the standard elements. The additional scenarios were initiated by the *Physical Link* (10.9) and the *Interface* (11.2). There was also a difference regarding privacy-related threats. Group G_{CPDFD} found on average 5.0 scenarios while the G_{DFD} found less than 1 scenario. The difference is not statistically significant, but can be explained by the fact that G_{CPDFD} found 3.6 scenarios with the *Physical Link* and 1.2 scenarios with the *Interface*. This effect could also be seen in the hardware model. The median found 29.5 and 11.5 scenarios for G_{CPDFD} and G_{DFD}, respectively. More than half of the scenarios of the former were based on the *Physical Link* (6) and *Interface* (10.5). It was also analysed how participants modelled specific components. For example, almost all of group G_{CPDFD} modelled the microphone as *Physical Link*. In contrast, only 42% of G_{DFD} modelled it. They were indecisive which element to use, as 3 participants modelled it as *Process* and 2 as *External Entity*. The interfaces such as JTAG and Bluetooth were modelled by about 90% of G_{CPDFD} while these were modelled by only one third or less of G_{DFD}. This means that the existence of the two new elements leads to such components being modelled at all. The two elements do not make the technique more difficult, as the questionnaire showed. In the survey, the majority of 75% agreed that each *Physical Link* and *Interface* provide added value and 73% agreed that they contribute to more accurate modelling of IoT devices (see subsection VI-B). In the interviews, the elements were generally seen positive. One security expert confirmed the problems with modelling certain hardware components without the new elements. The *Physical Link* was said to be good as data is generated there and actuators can influence the environment and the *Interface* was conceived to be important as debug interfaces are often overlooked, among others. However, it was noted a few times that the differentiation between both elements is not clear and that they could be summarised in one element. In summary, there are many arguments in favour of introducing the two new elements. The *Physical Link* leads to several additional scenarios, as seen in the experimental study, although not as many as for the *Interface*. The element can be used to represent sensors, which in turn have mainly contributed to the identification of threats against privacy. This added value was also endorsed by the survey participants. The results thus suggest that the *Physical Link* improves the threat identification. The situation is similar for the *Interface*. Quantitatively, even more scenarios could be identified with this element. The importance of interfaces was highlighted by students as well as security experts and engineers. Therefore, the results suggest the *Interface* improves the threat identification as well.

C. Limitations

Some limitations must be taken into account in the reported results. The HWD produced dozens of attack scenarios, but several of them may also be found in DFDs, creating overlaps. Consequently, the number of additional scenarios contributed by the HWD is lower than the raw count suggests. The quantitative effects of the two new elements must also be considered with caution. In the experimental study, participants of G_{CPDFD} had predefined stencils for the *Physical Link*, e.g., microphone and loudspeaker, and for the *Interface*, e.g., JTAG and Bluetooth. The list of dozens of stencils had to be searched through first, but these increase the likelihood that such components will be modelled. The modelling then automatically led to further scenarios. For a more detailed examination of the effects of the two elements, further groups would have been necessary, which would not have had predefined stencils available.

VIII. Conclusions and Future Work

The many security incidents in the IoT highlight the ongoing need for improved security. Although numerous researchers

address IoT security, systematic approaches are often lacking. This work introduces a modelling technique tailored to IoT devices to simplify modelling and enhance threat identification, helping manufacturers address weaknesses early in the product development cycle and support security by design. The proposed CPDFD technique considers the special circumstances of IoT devices. Due to their use in critical applications as well as the high accessibility, the creation of a hardware model was proposed. In addition, due to the interaction with the environment and the high connectivity of IoT devices, the two new elements *Physical Link* and *Interface* were introduced. Across two studies, an experimental quantitative comparison was conducted and the perspectives of security experts and engineers were collected. In summary, all three changes of the CPDFD technique indicate improvements compared to the standard DFD. The technique enables more detailed modelling of IoT devices, which in turn leads to more identified attack scenarios, as the results demonstrated. This suggests that CPDFDs improve the identification of threats. In the future, the better integration of HWDs and DFDs will be addressed to exploit synergy effects and improve the identification of attack scenarios and countermeasures.

REFERENCES

[1] SonicWall, *Annual number of Internet of Things (IoT) malware attacks worldwide from 2018 to 2022*, Statista, 2023. Accessed: 2026-03-14. [Online]. Available: https://www.statista.com/statistics/1377569/worldwide-annual-internet-of-things-attacks/.

[2] D. E. Whitehead, K. Owens, D. Gammel, and J. Smith, 'Ukraine cyber-induced power outage: Analysis and practical mitigation strategies', in *2017 70th Annual Conference for Protective Relay Engineers (CPRE)*, 2017, pp. 1–8. DOI: 10.1109/CPRE.2017.8090056.

[3] D. Truxius et al., 'Cyber Security Review of Network-Connected Medical Devices', German Federal Office for Information Security (BSI), Tech. Rep., Dec. 2020. Accessed: 2026-03-14. [Online]. Available: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/DigitaleGesellschaft/ManiMed_Abschlussbericht_EN.pdf?__blob=publicationFile&v=1.

[4] D. Comobo, *How I got access to 25+ Tesla's around the world. By accident. And curiosity.* Jan. 2022. Accessed: 2026-03-14. [Online]. Available: https://medium.com/@david_colombo/how-i-got-access-to-25-teslas-around-the-world-by-accident-and-curiosity-8b9ef040a028.

[5] C. R. Lab, *2025: Unpacking the Tesla Wall Connector exploit chain and its broader cybersecurity implication*, Aug. 2025. Accessed: 2026-03-14. [Online]. Available: https://vicone.com/blog/from-pwn2own-automotive-2025-unpacking-the-tesla-wall-connector-exploit-chain-and-its-broader-cybersecurity-implication.

[6] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman, 'Green Lights Forever: Analyzing the Security of Traffic Infrastructure', in *8th USENIX Workshop on Offensive Technologies (WOOT 14)*, San Diego, CA: USENIX Association, Aug. 2014. Accessed: 2026-03-14. [Online]. Available: https://www.usenix.org/conference/woot14/workshop-program/presentation/ghena.

[7] D. Giese, 'Reverse engineering and hacking Ecovacs robots', DEFCON 32, 11th Aug. 2024. Accessed: 2026-03-14. [Online]. Available: https://dontvacuum.me/talks/DEFCON32/DEFCON32_reveng_hacking_ecovacs_robots.pdf.

[8] B. Dorsemaine, J.-P. Gaulier, J.-P. Wary, N. Kheir, and P. Urien, 'Internet of Things: A Definition & Taxonomy', in *2015 9th International Conference on Next Generation Mobile Applications, Services and Technologies*, Cambridge, United Kingdom: IEEE, Sep. 2015, pp. 72–77, ISBN: 978-1-4799-8660-6. DOI: 10.1109/NGMAST.2015.71. Accessed: 2026-03-14. [Online]. Available: http://ieeexplore.ieee.org/document/7373221/.

[9] K. Rose, S. Eldridge, and L. Chapin, 'The internet of things: An overview', *The internet society (ISOC)*, vol. 80, pp. 1–50, 2015, Publisher: Reston, VA.

[10] M. A. Iqbal, S. Hussain, X. Huanlai, and M. A. Imran, *Enabling the internet of things: fundamentals, design, and applications* (Wiley - IEE), First edition. Hoboken, NJ: Wiley, 2020, ISBN: 978-1-119-70125-5.

[11] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson, 'The industrial internet of things (IIoT): An analysis framework', *Computers in Industry*, vol. 101, pp. 1–12, Oct. 2018, ISSN: 01663615. DOI: 10.1016/j.compind.2018.04.015. Accessed: 2026-03-14. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0166361517307285.

[12] S. Liebl et al., 'Analyzing the Attack Surface and Threats of Industrial Internet of Things Devices', *International Journal on Advances in Security*, 1 & 2, vol. 14, pp. 59–70, Dec. 2021, ISSN: 1942-2636. Accessed: 2026-03-14. [Online]. Available: https://www.thinkmind.org/index.php?view=article&articleid=sec_v14_n12_2021_6.

[13] T. DeMarco, 'Structure Analysis and System Specification', in *Pioneers and Their Contributions to Software Engineering: sd&m Conference on Software Pioneers, Bonn, June 28/29, 2001, Original Historic Contributions*, M. Broy and E. Denert, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 255–288, ISBN: 978-3-642-48354-7. DOI: 10.1007/978-3-642-48354-7_9. Accessed: 2026-03-14. [Online]. Available: https://doi.org/10.1007/978-3-642-48354-7_9.

[14] A. Shostack, *Threat modeling: designing for security*. Indianapolis, IN: Wiley, 2014, ISBN: 978-1-118-80999-0.

[15] L. Kohnfelder and P. Garg, 'The threats to our products', Microsoft, Tech. Rep., Apr. 1999.

[16] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, 'A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements', *Requirements Engineering*, vol. 16, no. 1, pp. 3–32, Mar. 2011, ISSN: 0947-3602, 1432-010X. DOI: 10.1007/s00766-010-0115-7. Accessed: 2026-03-14. [Online]. Available: http://link.springer.com/10.1007/s00766-010-0115-7.

[17] Y. Harbi, Z. Aliouat, S. Harous, A. Bentaleb, and A. Refoufi, 'A Review of Security in Internet of Things', *Wireless Personal Communications*, vol. 108, no. 1, pp. 325–344, Sep. 2019, ISSN: 0929-6212, 1572-834X. DOI: 10.1007/s11277-019-06405-y. Accessed: 2026-03-14. [Online]. Available: http://link.springer.com/10.1007/s11277-019-06405-y.

[18] F. A. Alaba, M. Othman, I. A. T. Hashem, and F. Alotaibi, 'Internet of Things security: A survey', *Journal of Network and Computer Applications*, vol. 88, pp. 10–28, Jun. 2017, ISSN: 10848045. DOI: 10.1016/j.jnca.2017.04.002. Accessed: 2026-03-14. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1084804517301455.

[19] E. Leloglu, 'A Review of Security Concerns in Internet of Things', *Journal of Computer and Communications*, vol. 05, no. 01, pp. 121–136, 2017, ISSN: 2327-5219, 2327-5227. DOI: 10.4236/jcc.2017.51010. Accessed: 2026-03-14. [Online]. Available: http://www.scirp.org/journal/doi.aspx?DOI=10.4236/jcc.2017.51010.

[20] O. I. Abiodun, E. O. Abiodun, M. Alawida, R. S. Alkhawaldeh, and H. Arshad, 'A Review on the Security of the Internet of Things: Challenges and Solutions', *Wireless Personal Communications*, vol. 119, no. 3, pp. 2603–2637, Aug. 2021, ISSN: 0929-6212, 1572-834X. DOI: 10.1007/s11277-021-08348-9. Accessed: 2026-03-14. [Online]. Available: https://link.springer.com/10.1007/s11277-021-08348-9.

[21] M. Scott, 'A Survey Study of Common Security Failures and Mitigations for the Internet of Things (IoT)', *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 15, no. 2, pp. 9–15, Jun. 2023. Accessed: 2026-03-14. [Online]. Available: https://jtec.utem.edu.my/jtec/article/view/6259.

[22] O. Arias, J. Wurm, K. Hoang, and Y. Jin, 'Privacy and Security in Internet of Things and Wearable Devices', *IEEE Transactions on Multi-Scale Computing Systems*, vol. 1, no. 2, pp. 99–109, 2015. DOI: 10.1109/TMSCS.2015.2498605.

[23] C. Kolias, A. Stavrou, J. Voas, I. Bojanova, and R. Kuhn, 'Learning Internet-of-Things Security "Hands-On"', *IEEE Security & Privacy*, vol. 14, no. 1, pp. 37–46, 2016. DOI: 10.1109/MSP.2016.4.

[24] J. Wurm, K. Hoang, O. Arias, A.-R. Sadeghi, and Y. Jin, 'Security analysis on consumer and industrial IoT devices', in *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2016, pp. 519–524. DOI: 10.1109/ASPDAC.2016.7428064.

[25] A. W. Atamli and A. Martin, 'Threat-Based Security Analysis for the Internet of Things', in *2014 International Workshop on Secure Internet of Things*, 2014, pp. 35–43. DOI: 10.1109/SIoT.2014.10.

[26] S. Iskhakov, A. Shelupanov, and A. Mitsel, 'Internet of Things: Security of Embedded Devices', in *2018 3rd Russian-Pacific Conference on Computer Technology and Applications (RPC)*, 2018, pp. 1–4. DOI: 10.1109/RPC.2018.8482148.

[27] V. Casola, A. De Benedictis, M. Rak, and U. Villano, 'Toward the automation of threat modeling and risk assessment in IoT systems', *Internet of Things*, vol. 7, p. 100 056, Sep. 2019. DOI: 10.1016/j.iot.2019.100056. Accessed: 2026-03-14.

[28] R. Khan, K. McLaughlin, D. Laverty, and S. Sezer, 'STRIDE-based threat modeling for cyber-physical systems', in *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2017, pp. 1–6. DOI: 10.1109/ISGTEurope.2017.8260283.

[29] M. Hamad, M. Nolte, and V. Prevelakis, 'Towards Comprehensive Threat Modeling for Vehicles', *Publications Institute of Computer and Network Engineering*, 2016. DOI: 10.24355/dbbs.084-201806251532-0.

[30] D. Meyer, J. Haase, M. Eckert, and B. Klauer, 'A threat-model for building and home automation', in *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, 2016, pp. 860–866. DOI: 10.1109/INDIN.2016.7819280.

[31] M. R. A. Asif, K. F. Hasan, M. Z. Islam, and R. Khondoker, 'STRIDE-based Cyber Security Threat Modeling for IoT-enabled Precision Agriculture Systems', in *2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, 2021, pp. 1–6. DOI: 10.1109/STI53101.2021.9732597.

[32] A. Omotosho, B. A. Haruna, and O. M. Olaniyi, 'Threat Modeling of Internet of Things Health Devices', *Journal of Applied Security Research*, vol. 14, no. 1, pp. 106–121, Jan. 2019. DOI: 10.1080/19361610.2019.1545278. Accessed: 2026-03-14. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/19361610.2019.1545278.

[33] B. J. Berger, K. Sohr, and R. Koschke, 'Automatically Extracting Threats from Extended Data Flow Diagrams', in *Engineering Secure Software and Systems*, J. Caballero, E. Bodden, and E. Athanasopoulos, Eds., vol. 9639, Cham: Springer International Publishing, 2016, pp. 56–71, ISBN: 978-3-319-30805-0. DOI: 10.1007/978-3-319-30806-7_4. Accessed: 2026-03-14. [Online]. Available: http://link.springer.com/10.1007/978-3-319-30806-7_4.

[34] L. Sion, K. Yskout, D. Van Landuyt, and W. Joosen, 'Solution-aware data flow diagrams for security threat modeling', in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, Pau France: ACM, Apr. 2018, pp. 1425–1432, ISBN: 978-1-4503-5191-1. DOI: 10.1145/3167132.3167285. Accessed: 2026-03-14. [Online]. Available: https://dl.acm.org/doi/10.1145/3167132.3167285.

[35] Fraunhofer Institute for Applied and Integrated Security (AISEC), 'A Study on Hardware Attacks against Microcontrollers', German Federal Office for Information Security (BSI), Tech. Rep., Mar. 2023.

[36] D. D. Gajski, S. Abdi, A. Gerstlauer, and G. Schirner, *Embedded System Design: Modeling, Synthesis and Verification*. Boston, MA: Springer US, 2009. DOI: 10.1007/978-1-4419-0504-8. Accessed: 2026-03-14. [Online]. Available: https://link.springer.com/10.1007/978-1-4419-0504-8.

[37] E. Bochniewicz et al., *Playbook for Threat Modeling Medical Devices*. MITRE Corporation and Medical Device Innovation Consortium (MDIC).

[38] M. Wolf, 'Combining safety and security threat modeling to improve automotive penetration testing', 2019, Publisher: Universität Ulm. DOI: 10.18725/OPARU-13062. Accessed: 2026-03-14. [Online]. Available: https://oparu.uni-ulm.de/xmlui/handle/123456789/13119.

[39] electrouser865, *Jaimico: The New I-health Care Companion Robot*. Accessed: 2026-03-14. [Online]. Available: https://www.electromaker.io/project/view/jaimico-the-new-i-health-care-companion-robot.

[40] emgarde, 'emgarde | TTModeler Pro', Accessed: 2026-03-14. [Online]. Available: https://emgarde.de.

[41] S. Liebl, *TTModeler*. Accessed: 2026-03-14. [Online]. Available: https://github.com/SecSimon/TTM.

[42] International Organization for Standardization and International Electrotechnical Commission, *ISO/IEC 25022:2016: Systems and Software Engineering - Systems and Software Quality Requirements and Evaluation (SQuaRE) - Measure of Quality in Use* (International standard). ISO, 2016. Accessed: 2026-03-14. [Online]. Available: https://books.google.de/books?id=Wzc1ygEACAAJ.

[43] J. Ziegeldorf, O. Morchon, and K. Wehrle, 'Privacy in the internet of things: Threats and challenges', *Security and Communication Networks*, vol. 7, Dec. 2014. DOI: 10.1002/sec.795.

[44] S. H. Simpson, 'Creating a Data Analysis Plan: What to Consider When Choosing Statistics for a Study.', *The Canadian journal of hospital pharmacy*, vol. 68 4, pp. 311–7, 2015.

[45] R. McGill, J. W. Tukey, and W. A. Larsen, 'Variations of box plots', *The American Statistician*, vol. 32, no. 1, pp. 12–16, 1978, Publisher: Taylor & Francis Group.

[46] J. Cohen, *Statistical power analysis for the behavioral sciences*, 2nd ed. Hillsdale, N.J: L. Erlbaum Associates, 1988, ISBN: 978-0-8058-0283-2.

[47] J. Hartung, G. Knapp, and B. K. Sinha, *Statistical Meta-Analysis with Applications* (Wiley Series in Probability and Statistics). Hoboken, NJ, USA: John Wiley & Sons, Inc., Jul. 2008, ISBN: 978-0-470-29089-7. DOI: 10.1002/9780470386347.

[48] Foreign, Commonwealth & Development Office, *Small to medium sized enterprise (SME) action plan*, May 2023. Accessed: 2026-03-14. [Online]. Available: https://www.gov.uk/government/publications/fcdo-small-to-medium-sized-enterprise-sme-action-plan/small-to-medium-sized-enterprise-sme-action-plan.

[49] R. Likert, 'A technique for the measurement of attitudes.', *Archives of Psychology*, vol. 22 140, pp. 55–55, 1932.