TU WIEN Informatics

IoT Guard: Usable Transparency and Control Over Smart Home IoT Devices

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Software Engineering/Internet Computing

eingereicht von

Stefan Victora, Bsc
Matrikelnummer 01026993

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Privatdoz. Mag.rer.soc.oec. Dipl.-Ing. Dr.techn. Edgar Weippl

Wien, 12. Mai 2020

Stefan Victora	Edgar Weippl

TU WIEN Informatics

IoT Guard: Usable Transparency and Control Over Smart Home IoT Devices

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Software Engineering/Internet Computing

by

Stefan Victora, Bsc
Registration Number 01026993

to the Faculty of Informatics

at the TU Wien

Advisor: Privatdoz. Mag.rer.soc.oec. Dipl.-Ing. Dr.techn. Edgar Weippl

Vienna, 12th May, 2020

Stefan Victora	Edgar Weippl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.at

Erklärung zur Verfassung der Arbeit

Stefan Victora, Bsc

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 12. Mai 2020

Stefan Victora

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my family for their endless support on the strenuous journey of this thesis. In the same way, I would like to thank my friends for their moral support, especially in times when I had more than enough of transcribing and analysing the seemingly endless interviews. Nevertheless, I am very grateful to all participants of my study for providing their time and invaluable input that led to the results of this thesis. Finally, I would also like to thank my supervisors for their patience and valuable feedback, which significantly improved the final version of this work.

Although this thesis required a *lot* more time and effort than I had expected, I am still glad that I chose such a timely topic, and hope that the results will be a valuable contribution not only to academic research but also to the actual users of smart devices – I think it's long overdue to finally gain insight into our devices' communication behaviour, and empower even non-experts to take control of their smart device-related privacy.

Kurzfassung

Die Zahl der Smart-Home Internet of Things (IoT)-Geräte wächst kontinuierlich, wodurch zwar neue Automatisierungen und Bequemlichkeiten ermöglicht werden, die Geräte aber zugleich in der Lage sind, immer mehr Daten aus noch mehr Bereichen unseres Lebens zu sammeln. Frühere Arbeiten haben zusätzlich gezeigt, dass Benutzer wenig bis gar kein Bewusstsein für das Kommunikationsverhalten ihrer Geräte haben, wodurch sie ihre Datenschutzentscheidungen oft auf unvollständige mentale Modelle stützen. Dies erhöht das Risiko für eine Datenschutzverletzung, sollten sich diese nicht wie erwartet verhalten. Bestehende Forschung hat deshalb bereits Transparenz und Kontrolle als zwei Grundvoraussetzungen für effektive Smart-Home-Datenschutzmechanismen identifiziert. Bis jetzt wurde jedoch nur unzureichend untersucht, wie diese in einer benutzerfreundlichen Art und Weise bereitgestellt werden können.

Diese Arbeit aus den Forschungsgebieten der Usable Security und Netzwerk-Forensik verwendet deshalb eine Designstudienmethodik, um Benutzer direkt in den Designprozess einzubinden, und untersucht, wie der ausgehende Netzwerkverkehr eines Haushalts in einer brauchbaren Weise visualisiert werden kann, sodass selbst Nichtexperten selbständig potenziell unerwünschtes Verhalten identifizieren und Datenschutzentscheidungen treffen können. Um diese Frage zu beantworten, wurden 12 ausführliche semi-strukturierte Interviews mit ergänzenden Participatory Design (PD)-Aufgaben durchgeführt, in denen die Teilnehmer zusätzlich Low-Fidelity Skizzen ihrer gewünschten Datenabstraktion und Interaktion erstellten. Aus diesen qualitativen Ergebnissen wurden mehrere Design-Richtlinien abgeleitet, darunter die Verwendung von kontaktierten Unternehmen als primäre Datenabstraktion für das Kommunikationsverhalten der Geräte, die es den Benutzern erlaubt, ihre vorhandenen mentalen Modelle wiederzuverwenden. Zur Überraschung des Autors wollten alle Teilnehmer selbst auf Grundlage ihrer persönlichen Präferenz und zusätzlicher Informationen über Unternehmen entscheiden, was blockiert werden soll. Weitere Richtlinien umfassen zum Beispiel eine einfache Übersicht und eine zeitbasierte Visualisierung. Um deren Wirksamkeit zu validieren, wurde ein Werkzeug implementiert, das die gewünschten Informationen automatisch aus dem Netzwerkverkehr der Geräte extrahieren kann. Dessen Visualisierung wurde dabei iterativ durch sechs Usability-Tests verfeinert und schließlich durch reale Einsätze validiert. Die Ergebnisse bieten erste Indizien, dass die angewandten Design-Richtlinien effektiv den gewünschten Einblick und Kontrolle bieten können. Abschließend werden mehrere vielversprechende Richtungen für zukünftige Forschungen und Anwendungen vorgestellt.

Abstract

There is a growing number of smart home Internet of Things (IoT) devices that enable new automations and conveniences, but at the same time are able to collect more and more data on even more aspects of our lives. Moreover, previous work has shown that consumers have little to no awareness of their devices' communication behaviour, so they often base their privacy trade-offs on incomplete mental models. This increases their risk for privacy violations if their devices don't behave like expected. Related work has therefore already identified transparency and control as two primary requirements for effective smart home privacy mechanisms. However, little research has been done on how to actually provide them in a usable way, with existing tools using unsuitable abstractions that require too much effort.

This thesis from the research fields of usable security and network forensic thus uses a design study methodology to directly involve users in the design process, and investigates how a household's outgoing network traffic can be visualised in a usable way, so that non-expert users can independently identify potentially unwanted behaviour and make privacy-related decisions on their own. To answer this question, 12 semi-structured in-depth interviews with Participatory Design (PD) exercises were conducted, in which participants created low-fidelity sketches of their desired data abstraction and interaction. From these qualitative results several design guidelines were derived, including the use of contacted organisations as a primary data abstraction for devices' communication behaviour, which allows users to reuse their existing mental models. To the author's surprise, all participants wanted to decide for themselves what should be blocked, based on their personal preferences and additional information about organisations such as their business model. Potentially unwanted organisations should be explicitly highlighted. Further guidelines include, for example, a quick and simple overview, and a time-based visualisation in order to detect and highlight changes in behaviour. To validate their effectiveness, a tool has been implemented that can automatically extract the desired information from devices' network traffic with the help of several external organisation sources. Its visualisation was iteratively refined through six usability tests and finally validated through real-world deployments with target users. The results offer suggestive evidence that the applied design guidelines can effectively provide the desired insight and control, with the created visualisation improving the current state of the art, while requiring zero configuration overhead. Finally, several promising directions for future research and applications are presented.

Contents

CHAPTER 1

Introduction

There has been a continuous growth in the number of consumer Internet of Things (IoT) devices and forecasts by, for example, Gartner [52] still predict this trend to continue, with 13.5 billion devices and a spending of 1.5 trillion US dollars expected by the year 2020. While these devices might enable new automations and conveniences in our homes, this vision for ubiquitous connectivity also leads to an increase in data collection in even more aspects of our lives, with devices fading further into the background of our homes [66, 135]. Research has already shown that consumers have little to no awareness of their devices' communication behaviour [39, 87, 150], causing them to make ill-informed trade-offs between privacy and convenience [170], which increases their risk for privacy violations, if devices do not conform to their expectations [5, 79, 153].

Additionally, it's becoming increasingly difficult for consumers to even buy non-smart versions of certain devices, like TVs without Internet functionality [97, 135]. While consumers value their devices' smart features [53], previous work has already shown that they are concerned about the occurring data collection [39, 87, 98], especially for purposes not related to devices' main functionality, like, for example, targeted advertising [5, 97, 172]. However, recent news reports showed that, for example, smart TV manufacturers already calculate "post-purchase monetization" into their price model, mining users' behaviour for their profit [119]. And there are reasons to believe that such monetization practice, already known from the Web, will be applied more widely to the coming generations of smart devices [66], especially because these devices can collect personal information in real-time directly from inside consumers' homes [135].

Recent studies have therefore already identified *data transparency and control* as two primary requirements for effective privacy mechanisms in smart homes [150, 169, 172]. However, existing research has not yet explored how to actually provide consumers with such transparency and control in a usable way. While Yao et al. [169] included users in a co-design activity, they did not focus on designing user interfaces, which led participants to, for example, imagine purpose-built hardware. Even though a current research project

at Princeton University focuses on visualising network traffic, they likewise did not involve users in their design process [67]. Due to this lack of usable tools, Tabassum et al. [150] therefore called for research on novel, easy to use security and privacy tools, which, according to Hong [66], would also help regulatory agencies and journalists to make the public aware of a potentially unwanted device behaviour [36, 47]. Previous research by the author similarly identified this lack of tools; this thesis hence investigated the following research question using a design study methodology, involving 12 participants in semi-structured interviews and participatory design exercises:

> How can a household's outgoing network traffic be visualised in a usable way, giving non-expert users insight into the communication behaviour of their smart devices, enabling them to independently identify potentially unwanted behaviour and make privacy-related decisions on their own?

And makes the following contributions to the research field of usable privacy and security:

- Supports and systematises the need for transparency and control by replicating and summarising existing qualitative results, and extends the state of the art by providing further details on how users envision a usable control over their smart devices' network traffic.
- Presents the results of a participatory design exercise, where participants created sketches on how they imagined a usable visualisation of smart devices' communication behaviour, including details on how to identify and highlight potentially unwanted communication.
- Implements a tool based on the gathered design requirements, which provides usable control by being able to block whole organisations and their subsidiaries, and evaluates it in real-world scenarios by the author and target users.

The remaining thesis is split into qualitative results and implementation details, with Chapter 2 at first describing the procedure of the conducted design study in detail. In the first part, Chapter 3 strengthens and summarises the need for transparency and control; Chapter 4 describes how to identify potentially unwanted communication behaviour using the theory of Contextual Integrity (CI), in addition to participants' requirements for usable control; and Chapter 5 presents participants' sketches created in the participatory design using the identified data abstractions. In the second part, Chapter 6 details how the tool of this thesis extracted the required information from low-level packet data, what challenges it faced, and how the data was transformed into higher abstractions; Chapter 7 details the visual design and interactions of the created tool; and Chapter 8 evaluates the tool and discusses the results and its limitations. Finally, Chapter 9 concludes this thesis with a list of design recommendations and future work.

Threat Model

The focus of this thesis is on visualising external entities contacted by smart home devices, as participants were mostly concerned about sensitive data being sent to possibly unknown or unwanted organisations. Providing transparency and control over such behaviour enables them to act on potential violations of privacy expectations. Additionally, a tool that analyses the network traffic is able to cover the wide array of different smart home devices already on the market, without having to install custom software on each of these devices, which Hong [66] highlighted as perhaps the only scalable approach. Throughout this thesis, the general term "IoT" is used by participants as a synonym for the consumer smart home and its devices, whereas a "smart device" is considered a "context-aware electronic device capable of performing autonomous computing and connecting to other devices wire or wirelessly for data exchange." [146]

Due to the same reasons as Zheng et al. [172], this thesis does not discuss in-home privacy threats caused by data sharing between devices, or in-home adversaries like domestic abuse [15], as none of the participants voiced any concerns during the study. However, Zheng et al. note that this lack of participants' concern is not representative of smart device users, and the author of this thesis is aware that a visualisation tool for smart devices' communication behaviour might cause unintended negative consequences in domestic abuse cases, since it can increase the "asymmetry of power" of the abuser through invasive transparency and control over a household's connected devices [15]. The related Princeton IoT Inspector research [67], for example, tried to reduce such privacy violations by requiring the input of a device's Media Access Control (MAC) address in order to view its details, but they note that this only increases the barrier for malicious users without preventing such access. For non-adversarial households Zeng et al. [171] further reported that social norms within the home were more effective than software restrictions to prevent such abusive usage in their study, but added that there is a critical need for further investigation of adversarial situations, which includes research on the impact of such tools and how abusive usage can be mitigated. The remainder of this thesis focuses on non-adversarial households, and potentially unwanted external entities contacted by smart home devices.

CHAPTER 2

Methodology

To create a tool that provides the desired transparency and control in a usable way, a *design study* inspired methodology was used, in which researchers "analyse a specific real-world problem faced by domain experts, design a visualisation system that supports solving this problem, validate the design, and reflect about lessons learned in order to refine visualisation design guidelines," as Sedlmair et al. defined [142]. An important aspect of a design study is the close collaboration with target users, which in this thesis are smart home users with low network expertise, placing the work in the research fields of Human-Computer Interaction (HCI), Usable Security, and Information Visualisation (InfoVis). Oulasvirta et al. [116] similarly defined a HCI research problem as "a stated lack of understanding about some phenomenon in human use of computing, or stated inability to construct interactive technology to address that phenomenon for desired ends," with the research of this thesis addressing an *empirical-constructive* problem [116]. Likewise, Lazar et al. [88] defined such research as a combination of both *empirical* and *system research*.

The following sections focus on the core stages of a design study, as shown in Figure 2.1: Section 2.1 discusses the data gathering procedure and analysis, covering both the "discover" and "design" stages; Section 2.2 outlines how topics for the implementation where selected, how the design was refined through usability tests, and how it was evaluated by being deployed in real-world situations; Section 2.3 concludes this chapter by discussing the limitations of this study.

2.1 Discover and Design

The first stage of the core phase, "discover," is equivalent to a *requirements analysis* known from software engineering and aims to understand the "practices, needs, problems, and requirements" of the target users and their domain, which are non-expert users and their smart homes in this thesis [142]. This requires talking directly with the end

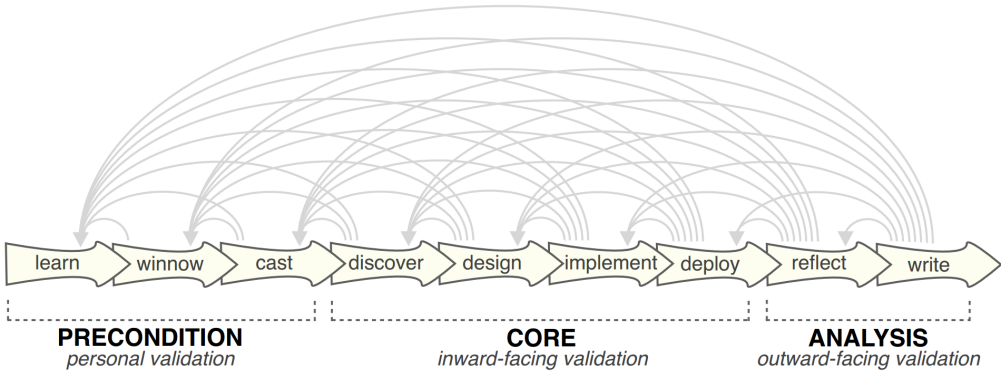


Figure 2.1: Design study stages, which inspired the methodology of this thesis. While the process is linear, the stages can overlap and repeat to refine initial ideas [142].

users, with *semi-structured interviews* considered a suitable method, as their flexible and open-ended structure allows participants to explore a topic in depth, which would not have been possible using more formal methods like surveys [88].

However, research showed that most users have difficulty introspecting about their needs when just being interviewed. Lazar et al. [88], among others, therefore proposed to combine interviews with some form of observation. In this thesis a *Participatory Design (PD)* exercise was used, which also bridged the gap to the "design" stage by including participants directly in the design process, which revealed some implicit knowledge not uncovered through the interviews. Although it has been shown before and again in this thesis that participants in a PD exercise mostly come up with designs inspired by systems they already know [161], it nevertheless highlighted design elements that were most important to them, and strengthened the qualitative results of the interviews.

2.1.1 Selection of Participants

Compared to surveys, semi-structured interviews are more challenging and time-consuming to conduct, which restricts the number of participants, making a well-thought-out selection crucial [88]. For this study a total of 12 participants from 11 households were selected for the interviews, conducted in February 2018. Due to the high expenditure of time and easier access, participants were selected from the author's circle of friends and acquaintances living in Austria, until the author felt confident to have reached saturation, i.e., no substantially new information could be gathered. Table 2.1 gives an overview of the demographics of the participants, including their network expertise, which are based on informed guesses by the author. While the participants were known to the author beforehand, care was taken to select people with different age, gender, and most importantly, network experience, with everything below 4 considered non-expert in this thesis. However, the selection is not representative of the target group, which is not feasible for in-depth qualitative interviews, according to Lazar et al. [88], with further limitations of the selection discussed in detail in Section 2.3.

Additionally, three interviews were conducted as group interviews to further save time and reduce logistic challenges, which required some special considerations during the interviews and PD exercises that are discussed in detail in Section 2.1.3. The order of participants in Table 2.1 also reflects the order in which the interviews were conducted, with the colour coding highlighting participants that were interviewed together. On average, the interviews lasted 1.5 hours, with a maximum of 2.3 hours, totalling at 13.6 hours for 8 interview sessions. Three of the single interviews were conducted via Skype (P10, P9, P12), while the rest were done in person. All interviews were audio recorded with the permission of the participants.

For the later evaluation, a selection of the same participants (P3, P4, P6, P9, P12) and some additional members of their households were consulted. Details about the evaluation are discussed in more detail in Section 2.2.

2.1.2 Interview Procedure

The interviews started with an introduction, which contained a short summary of the problem description and the goal of this paper: *a user-friendly insight into the communication behaviour of consumer IoT / smart devices so that users feel informed about their behaviour*. Similar to related research, the word "privacy" was explicitly omitted in the introduction and in further questions in order not to bias any participants [39]. Additionally, a quick summary of the major research questions of the interviews were given to provide participants with an overview of what to expect:

- *Do you already use tools to gain insight into such communication behaviour?*
- *How big is your awareness / concern about smart device-related dangers and unwanted background activities? Do you have any concerns about purchasing or using certain smart devices? No concern is also a valid answer.*
- *What information about the occurring data transfer would help your feel more informed?*
- *How could a usable visualisation of this information look like for you? Try to imagine how you would visualise your ideas with a sketch.*

The last question also contained a hint about the concluding PD exercise in the form of a drawing task, which was intended to put users in the mindset of visualising their ideas. It was additionally mentioned that there are no right or wrong answers, and that any queries are allowed if something is not clear enough. Feedback to the interview questions led to two revisions of the script after the first interview (P1) and the fourth (P6), which simplified some questions, and made some optional, if they didn't contribute much insight or disturbed the flow of the interviews, which is a common approach for semi-structured interviews [88]. The following paragraphs discuss the structure and content of the interviews according to the final script, as found in the appendix.

Participant	Age, gender	Network expertise (1...low, 6...exceptional)	Smart devices*
P1	36, m	2	(Smart printer, Xbox)
P2	29, m	5	Amazon Echo, Android TV, (Nvidia Shield, Raspberry Pi)
P3	26, f	2	Smart audio receiver, (smart printer)
P4	27, m	4	Chromecast, smart audio receiver, (smart printer, Steam Link, Playstation)
P5	27, m	5	Smart TV, smart watch
P6	56, m	2	Two smart TVs, smart radio, (smart printer, e-reader)
P7	29, m	5	Chromecast, (e-reader, multiple gaming consoles)
P8	27, f	1	[same household as P7]
P9	31, m	3	Smart TV, smart audio receiver, Google Home Mini, Nest Protect, (Xbox)
P10	28, m	2	Chromecast, Fire TV Stick, always-on smart printer, (e-reader, Playstation)
P11	28, m	3	Chromecast, Fire TV Stick, smart TV, (Playstation)
P12	56, m	6	Fire TV Stick, smart TV, smart radio, smart Blu-ray player, car with smart functionality, smart plug, (Multiple Raspberry Pi, multiple gaming consoles)

Table 2.1: List of participants. *Devices in parentheses are connected devices that were additionally mentioned by participants, but do not conform to the definition of smart devices in Chapter 1.

Existing Tools

After basic demographic information and a list of smart devices in participants' homes were collected, the first topic, as mentioned above, examined any existing tools that provide insight into the communication behaviour of devices. As a previous investigation of the author and existing research have already indicated a lack of such tools for smart devices, it was explicitly mentioned that any tools, regardless of device or platform, are of interest, which helped uncover participants' familiarity, and most importantly, their gained insight, troubles, and desires for improvements with tools like Little Snitch and GlassWire. P1 even gave a live demo of his usage of Little Snitch, comparable to a contextual inquiry [88], which not only highlighted features that were most important to him but also which information he was missing (see Section 3.3). If no tools had been used before, they were asked whether they ever wanted to have such insight into the communication behaviour of their apps or devices, confirming if participants actually felt a lack of tools. Inspired by the gaps identified in existing research [53, 159], the participants were asked whether they also desired to assert control over the communication behaviour, and if so, why and in which situations. After they had a chance to answer for themselves, ad blockers were mentioned as an example of control they might already use or are familiar with.

Awareness and Concerns

The second topic was participants' awareness about smart device related dangers in order to elicit concerns about their usage of devices as well as potential purchase decisions. To start, participants were asked if they could recall news reports about smart device related dangers, and, if so, if they considered them reasonable or exaggerated, and if they worried their own devices might be affected as well. After letting them answer, a news report about the "My Friend Cayla" doll was given as an example, which was classified as an "illegal espionage apparatus" in Germany and had to be destroyed [115]. To further understand participants' worries, they were asked about their concerns of certain types of devices, and if they are generally worried about unknown communication occurring in the background. Participants were then prompted to discuss what device behaviour they would consider unwanted, including a question about types of data they would find particularly undesirable, and if the purpose of transmission would change their acceptance. And finally, they were asked if they had some implicit trust in certain manufacturers or organisations.

At the end of the second interview part, various other news reports were mentioned, including reports about smart TVs analysing the watched content [95], apps listening in the background for consumed media [96], and robot vacuum cleaners creating maps of users' homes with plans to share this information to third parties [7]. These news reports were deliberately used to illicit reactions about privacy and security concerns. While this might have biased participants to think about these topics, existing research and the results of this thesis have shown that users often have latent concerns, which can be brought up if privacy and security related information is provided [39] (see Section 3.1.3).

Furthermore, in order to elicit participants' design requirements and ideas for a usable visualisation of device behaviour, some awareness of existing unwanted behaviour was required. Some participants explicitly mentioned that it was advantageous to conclude with the design requirements and the PD exercise, as they had no concrete ideas straight from the beginning:

> As feedback to the questions: you've taken this conversation in a direction I didn't expect at first. But now that I think about it... I did have some ideas that I wouldn't have had if you had just asked "what do I think about it?" Or just: "tell me everything you would like to know about smart home and transparency." I would have never thought of these ideas. (P5)

Data Abstraction

The third part of the interview was about the type of information participants wanted to know in order to make them feel informed about unknown or potentially unwanted device behaviour. It was explicitly mentioned at the beginning that any ideas are allowed, even if they seem infeasible to implement with current technology, which especially helped participants with technical background to come up with new ideas. If they already used a tool, as discussed in the first part, they were asked again what kind of additional information they would like to have. The following questions then related to the granularity of information, as existing research has already shown that non-expert users don't care about low-level network details, and instead think about devices' communication behaviour in terms of performed activities like "watching Netflix" compared to "contacting `amazonaws.com`" [19]. Similarly, Van Kleek et al. [159] recently showed that participants preferred to know the contacted organisations instead of domains. However, in order not to bias participants' responses, the initial questions were kept very general and only asked about additional information participants desired about potentially unknown and cryptic domains. If participants themselves didn't come up with the idea of providing the organisation, they were asked whether they would value such information. While this might have influenced the result towards organisations, it was used to replicate the recent findings of Van Kleek et al., and to prompt the discussion about further relevant details about potentially unknown organisations, as their report found that an organisation's name alone is still not enough for participants to feel informed [159]. To conclude this part of the interview, participants were asked if there are any more relevant details about a device's communication behaviour that are of interest to them.

Participatory Design

Inspired by a PD [161] and conceptual mapping exercise [88], the final part of the interview consisted of a drawing exercise, where participants were asked to create sketches of how they imagine their key information to be visualised in a low-fidelity user interface. It was explicitly mentioned that not all aspects of the interface have to be drawn, just how they

imagine the information to be laid out, repeating that their ideas can be futuristic and don't have to be feasible with current technology. Furthermore, it was mentioned that they could seek help from the author at any time during the exercise if they feel stuck or don't know how to visualise their abstract ideas.

However, to the author's surprise, only P3 had troubles coming up with ideas for a visualisation, and also only in the beginning. All other participants quickly started drawing their ideas. On average the exercise lasted about 30 minutes, or one third of the interview, including the final discussion. To make participants' thought process visible during the design process, thinking aloud was encouraged, which however was only effectively applied in the one-on-one interviews, as the exercise was done in parallel for group interviews (see Section 2.1.3). During the exercise, some additional questions were asked, encouraging participants to think, for example, of ways how they could provide an overview by, for example, grouping certain information together. However, to the best of the author's knowledge, these questions did not influence their main design ideas in any noticeable way. To encourage further exploration of ideas, participants were also asked if they could imagine an alternative design to the one they sketched, which led P1 to create an alternative visualisation, and others to verbally discuss some possible variations.

Towards the end of the exercise, when participants largely completed their designs, additional questions about design aspects and interactions were asked, which led to some amendments to their sketches. For example, which information they would highlight or adjust to make the visualisation usable over a longer period of time, as previous research have noticed that participants were bothered about clutter after some time [19]. Finally, participants were asked how they would like to interact with their sketches, or how they imagine asserting control over the visualised communication behaviour, which inspired some participants to further add buttons and other control elements to their sketches. Concluding the interviews, participants were asked whether they had expected certain questions or topics that were not discussed, and whether they had additional stories about their usage of smart devices they would like to share, which led to further insight about their general mindset and concerns about smart devices.

2.1.3 Speciality of Group Interviews

As already mentioned, the group interviews required some special consideration of the interview and PD procedure. Due to the semi-structured nature of the interviews, the questions were not asked repeatedly for each participant, but instead were directed at the whole group, and whoever answered first naturally influenced the answers of the other participants. Furthermore, quite a lot of discussion happened between participants, uncovering new important topics for follow-up questions. While the author did not try to intervene in these open discussions, he encouraged participants that did not voice their opinions on some topics to speak up, without, however, forcing an answer. The dynamic of the group interviews were especially interesting in the groups with mixed network experience, like with P7 and P8, where P8 had only very low experience. While she didn't always have an opinion on the more technical questions, her insight and answers actually

influenced P7 to reflect on the feasibility of his ideas for less technical users. Similarly, P3 repeatedly interjected in the discussions between P2 and P4, that she and none of her friends would ever think or use such ideas, which caused the other participants to think about additional abstractions. These positive effects were also highlighted by Lazar et al. [88], provided participants are not in complete disagreement.

Furthermore, the group interviews also influenced the PD exercise, which were, as already mentioned, done in parallel for all participants. However, compared to the interviews, they each worked on their own, and, surprisingly, some formed a kind of competition between them, trying to prevent others from copying their ideas. This effectively prevented them from thinking aloud, which was encouraged in the single interviews. While participants solved their design exercises independently, the preceding group interview, however, naturally influenced their designs; P3 even explicitly stated that her sketch was mainly inspired by the ideas the other participants mentioned. Her sketch was, however, still vastly different from the others, underlining that the exercise helps to highlight the design elements considered most important to participants [161]. To compensate for the missing thinking aloud, each sketch was discussed individually after everyone finished, and participants were encouraged to explain their design ideas and trains of thought while creating their sketch. These discussions led to some insightful comparisons with the sketches of other participants, where, for example, P2 mentioned that his sketch would complement the one by P4 very well, and liked to have both visualisations in a single tool: "I think a combination of these designs is actually not so bad, if you can show the organisation-specific on the one hand and then here geographically, where it's really going" (P2).

2.1.4 Data Analysis

After the author felt he had reached saturation with the interviews, i.e., no fundamentally new information was gained by further interviews, they were transcribed and resulted in 205 pages with a total of 89.058 words. The interviews and scanned sketches were then coded using MAXQDA[1] and the open coding method, which is a form of emergent coding, i.e., coding without any previous taxonomies or theories in mind [88, 138]. Saldaña [138] described coding as an inductive, iterative, and highly subjective process, which tries to label and link the coded data in order to find patterns, categories, and themes over multiple coding cycles. To move from codes to categories and themes, a constant comparison approach was used, which compares new codes to the previously created ones, trying to find ways to relate and restructure them. To handle the expected large number of codes, analytic memos were used throughout the process, recording detailed descriptions and inclusion criteria for each code, as well as general thoughts, ideas, and to-do entries [138].

After saturation was reached, i.e., further coding cycles did not result in new categories or insight, frequency counts were used to select the most prominent codes and categories,

[1] https://www.maxqda.de/

which were written on index cards and arranged using the so-called "tabletop categories" method until a coherent story line emerged, linking the primary themes together [138]. This story line was used to structure the presentation of the initial study results and was further refined during the final write up. The final code book contains about 400 codes grouped by the following 7 top-level categories, which directly influenced that data chapters of this thesis: Chapter 3 covers *Attitude towards IoT devices*, *Existing monitoring*, and *Interest in a tool for IoT devices*; Chapter 4 *Target of transfer*, *Transferred data*, and *Control*; and Chapter 5 *Visualisation*.

A major challenge during the data analysis was the amount of data to process, which is supported by Saldaña [138], where he described qualitative data analysis as tedious and time-consuming work, requiring high concentration and creativity from the researcher, which caused the transcription and initial codings to range from 18.02.2018 to 07.06.2018, a total of about 15 weeks, which does not including the time for creating the final story line. Furthermore, as the interviews were conducted in German, the quotes used throughout this thesis were subsequently translated by the author.

2.2 Implement and Deploy

Finally, the stages "implement" and "deploy" conclude the core phase of the design study and, as their names suggest, are about implementing a software tool, which is validated by gathering feedback about its use by the target users [142]. The selection of topics to implement was directly inspired by the themes discovered through coding, with their feasibility intuitively judged by the author based on past experience in network forensic. This excluded, for example, the analysis of the type of transferred data, as existing research has shown that this would require an unencrypted view of the data stream, which is not easily accessible for self-contained smart devices (see Section 4.4).

2.2.1 Design Iterations

The core implementation stage lasted from 29.06.2018 to 5.11.2018 and included a continuous deployment in the author's home, continuously evaluating the tool in real-world conditions. The implemented visualisation (see Chapter 7) was inspired by participants' design requirements, while not directly copying any of their designs. To arrive at the final design, several iterations were created and validated through a total of six informal usability tests. In total five different participants (2×P6, P6w, P9, P4, P3) took part, with four already known from the interviews, and P6w being the wife of P6. To validate and improve the effectiveness of the applied abstractions and visual encodings, care was taken to only select non-expert participants (see Section 2.1.1), the target users of this research. According to Rubin et al. [136] the applied usability tests can be categorised as *assessment tests*, a formative evaluation of an already detailed but not finished design, with the goal to provide qualitative feedback about flaws and issues in the user interface [88].

The usability tests were conducted one-on-one in participants' homes, which provided real-world insight into the behaviour of their own devices, and, additionally, validated the tool's ability to run in different network environments. Due to the dynamic environment and informal character of the usability tests, no task lists for the participants were provided, instead, they were encouraged to explore the user interface on their own, using a thinking aloud approach, and only minimal intervention from the author. After participants finished their exploration, a concluding unstructured discussion about the tool and participants' understanding was conducted, in particular concerning the parts of the interface that were interpreted differently than expected. After each test, found flaws were fixed before the next round of usability tests, so the applied changes could be validated and participants could focus on any remaining flaws [166]. The gained insight and qualitative feedback gathered from the observations and the concluding discussions are presented in detail in Chapter 8. Due to the sensitivity of participants' network traffic, no analysis results were persisted or further analysed after the usability tests.

2.2.2 Deployment

After the last usability test, which only provided minimal new insight, the implementation stage was concluded by creating a self-contained image of the tool that could be run on a Raspberry Pi. This image was then shared with five users selected from the author's acquaintances, with the goal to collect anecdotal evidence of its utility, including reports about discovered unwanted behaviour of their devices. Three users, including P6 and P12, reported back and shared their gained insights and identified problems through informal in-person discussions as well as additional screenshots via email. Their feedback, in addition to the author's evaluations and deeper analysis of the reported tracking behaviour, are presented in Chapter 8.

2.3 Limitations

The last stages of a design study are about relating the results to the available literature, improving existing design guidelines as well as constructing a coherent story line [142]. Similar to related interview-based studies [150, 169, 172], the small sample size and non-representative selection of participants is a limiting factor for the results of this thesis. For example, the majority of participants were sceptical of smart devices and as a result only owned a limited number of them (see Table 2.1), which might have biased the results. However, this limitation is also a valuable contribution to existing research, which focused mainly on early adopter [150, 172], as participants' scepticism in this thesis provided useful insight into what information and control they desired in order to feel more comfortable with those devices, and further showed that such transparency and control is also of relevance for non-IoT devices like smartphones. Finally, due to time constraints of this thesis, no formal evaluations about in-the-wild usage or adoption rates of the implemented tool were conducted, which are, however, opportunities for future research.

Part I

Qualitative Results

CHAPTER 3

The Need for Usable Transparency and Control

Existing research has already identified transparency and control as essential requirements for effective privacy mechanisms in smart homes, however, insufficient attention has been paid to how these can be provided in a usable way. This chapter strengthens and summarises existing results, and argues why current tools don't fulfil participants' requirements, supporting the hypothesis that such transparency and control is actually needed. The next chapter extends these results and discusses in detail how to provide such transparency and control in a usable way.

3.1 The Need for Transparency

First of all, this section highlights the need for a usable transparency to not only improve users' privacy trade-offs but also raise their initial awareness of privacy problems that may be caused by smart devices.

3.1.1 Trade-Off Between Privacy and Convenience

A major theme identified through the interviews is that participants don't fully trust their devices or are generally concerned about unknown data collection, but some continue to use them anyway. P10 and P11, for example, strongly believe that their devices collect more data than the manufacturer is disclosing, like their Amazon Fire TV remote controls spying on them in the background: "I don't think there is even a debate, they're definitely doing that. [...] I'm sure they do more than they say. Absolutely. [P11: Less certainly not!] Yeah, I'm rather suspicious about that" (P10). However, despite their privacy concerns, they still continue to use their devices, as they see enough value as compensation:

> P11: I really don't trust the device [P10: No] [laughing] but it's quite convenient.
>
> P10: Yeah, exactly, that's what I'm thinking. Because typing is just really tedious, and the voice controls are actually quite good. [P11: Mhm] It's nice, but I also think that it's definitely running in the background [...] I know you have to press the button to activate it, but you can't be sure [P11: No] if it's really switched off if you're not pressing it

For the same reasons, P8 described how she would feel unwell or watched, owning, for example, an Intelligent Personal Assistant (IPA) like an Amazon Echo device, similarly reported by existing research [87, 98, 150]. Unlike P10 and P11, who see enough value as compensation, for her, the convenience is not worth her loss of privacy:

> [P7] already said – it would be quite nice, but I said, no, I don't like the fact that there is a microphone on all the time. [...] And I don't know, but I would feel really uncomfortable if I knew it had the ability to record all the time. And I wouldn't put it past it that this would really happen. (P8)

Her smartphone, on the other hand, provides so much value, that she is willing to accept that it may collect a lot of personal data and possibly share it with unwanted entities: "My phone is just so important to me, I need it a lot, *every* day – I'm aware that it collects a lot of data about me. But it adds so much value that I use it anyway" (P8). This trade-off between privacy and convenience is supported by Ghiglieri et al. [53], where a participant said: "I think that the advantages that I get when it's connected to the Internet outweigh the disadvantages." Similar mindsets were found by Zheng et al. [172]: "I like to say I'm thoughtful about these decisions, but very often the case is that something is really convenient so I'll do it anyways even if I do have some reservations about privacy" (P11 [172]).

3.1.2 Better Privacy Trade-Offs Through Transparency

The problem with this trade-off between privacy and convenience is that Lin et al. [92] showed that people often have quite simple and incomplete mental models about the occurring data collection on which they base their privacy decisions, which was also supported by the interviews. To improve such mental models and therefore users' privacy decisions, Malkin et al. [97] highlighted the importance of transparency – referring to the property of "visibility" [155] – about data collection and use, similarly supported by a variety of related research [111, 150, 169]. Van Kleek et al. [159], for example, reported that such transparency allows people to re-use their existing privacy preferences, which can vary greatly between people. Even legal frameworks like the European General Data Protection Regulation (GDPR) see such transparency as a basic requirement for users to exert their right of control over their privacy; without knowing what data is transferred to whom, no decision to mitigate privacy violations can be made [111, 135].

However, such transparency has to be provided in a usable form, or else it can't be used by the non-expert target users, as the GDPR emphasises: "to provide any information [...] relating to processing to the data subject in a concise, transparent, intelligible and easily accessible form, using clear and plain language" [23]; this includes the requirement for minimal overhead concerning configuration and maintenance [87, 169, 172]. Some participants have similarly highlighted the importance of usability for such tool: "But user-friendly! [...] I think that's important, so that the masses can use it" (P10). However, previous work has not specifically addressed how to provide such usable transparency; Chapter 4 is meant to close this gap.

3.1.3 Raising Awareness for Privacy Problems

Besides supporting privacy decisions, transparency is equally important to initially raise awareness for smart device related privacy issues, because, while all participants voiced some privacy concerns, some only did after the subject has been brought up (see Section 2.1.2): "Well, now I do... now I'm worried about it" (P3). "Actually, only just now... I haven't really... I didn't assume it did. But especially with Google Chromecast, it's rather obvious that it collects some kind of usage data" (P4). Such latent concerns were similarly observed by Lau et al. [87]: "Until we started having this conversation I never really thought of a smart speaker as having anything to do with privacy" (U08 [87]).

P10 explained his initial lack of awareness that he wouldn't actively think about his devices' behaviour if they provided their main functionality as expected: "But you don't really think about it, if you don't hear about it or actively investigate. It is what it is – as long as it works, it works and you use it. And everything else..." (P10), but highlighted the importance of increasing such awareness as an important research goal: "It would be good if more is done about that and people become more aware of what they are surrounding themselves with. [...] Because I do think that this may cause a lot of damage. Because if there is no awareness, why should I bother doing anything about it" (P10).

Finally, such improved awareness and transparency could also "level the playing field" for smaller and lesser known organisations that apply good privacy practices, as Emami-Naeini et al. [39] put it, and may, similarly, create better market incentives for manufacturers to improve the privacy and security of their products in order to avoid the negative publicity of violating users' expectations [134, 172].

3.2 The Need for Control

Even more important than transparency is having control over the communication behaviour of smart devices, and P1 and P7 even went so far as to say that they don't see the purpose of such visualisation tool unless it also provides control over the behaviour: "Yes, of course I would like to actively intervene. That's kind of the point, if you know something is not justified in your opinion, that you can stop it then" (P1).

3.2.1 Negative Effects of Transparency

Without control, the gained knowledge about unwanted behaviour would leave, for example, P7 even more worried and unwell in the presence of his devices:

> Because if you can't stop it, I wouldn't see the point... I mean, I would then somehow have to throw away or sell my device, if I'm not ok with that. [...] Otherwise, you just feel unwell all day long, if you're thinking – Uah, Alexa is constantly sending voice data, that's actually not what I wanted. (P7)

Van Kleek et al. [158] likewise reported such negative effects of transparency: "[Refine] also exposes how much you're not in control, if you know what I mean. So I come in here [lab], I come into this room, and I'm like... I trust my phone, I trust my apps, and now that you've shown me this, and I'm like... hang on!" (P19 [158]) A participant in the study of Karegar et al. [81] reacted in a similar way: "Now that I am informed about the data, what can I do about it? I need to react on it."

While Schaub et al. [140] found similar concerns among participants using browser-based privacy extensions, he more importantly showed that mechanisms to block unwanted behaviour can mitigate these concerns again: "I'm more concerned just from the standpoint that now I have more information and I have a little bit more control... Before it was uninformed concern and there wasn't much I can do about it. And now it's informed concern and I can do something about it" (P5 [140]).

3.2.2 Resignation About Lack of Control

Due to the current lack of tools for smart devices (see Section 3.3), some people have, however, already resigned to their lack of control over invasive data collection, thinking that it might be a necessary evil of smart functionality [97, 150, 158]: "Once I bought all these devices that was it. These functions come with these risks no matter what and I can't do anything about that. There are no third option. If you want the device you have to accept those risks, otherwise don't use it at all" (Id12 [150]). However, giving up on their smart features would cause some to feel severely limited in their lifestyle:

> In order to stop all this stuff, if the only thing I can do is uninstall these apps, then I would feel that my personal freedom had been severely restricted, limited. I couldn't function happily without these, I want them, so I'm going to risk whatever goes out there. But it's horrible. (P13 [158])

Not to mention that some users are aware that certain form of data collection is so prevalent that it may feel impossible to avoid:

> I would want to say that I'm annoyed by Google Ads and Facebook Analytics, but it's already so accepted that you can't do anything about it that I've

> almost accepted this learned helplessness, like there's no way to avoid that at all... I've grown so discouraged by my inability to change anything about it that I just think, you know what, I should maybe just go with it. (P12 [158])

"Your biggest control element," Schaub noted in [135], "is deciding which devices you place in your home and vetting them for good privacy practices," but added, "It's often difficult to find this information for consumer devices and take it into account in any kind of purchasing decisions." This lack of privacy and security-related information at the time of purchase is supported by Emami-Naeini et al. [39], where they proposed the creation of privacy labels that summarise the most important privacy and security properties in a usable way, which are discussed in more detail in Section 4.5.4. This lack of information also highlights the risk of switching to an equivalent product, as it may exhibit the same unwanted behaviour.

3.2.3 Resignation About Lack of Alternatives

In addition, there is further the risk that no suitable alternatives to privacy-invasive products might even exist, as participants of Van Kleek et al. [158] argued: "But something like [Bank], which we found was... surprisingly really bad – there is no substitute for that! You are a customer of [Bank]. I mean, it's hard to just go to another bank and say – hey I'm out. Just gotta suck it up and trust them" (P19 [158]). "You could use some lesser known dating app than Tinder, but you would never meet anyone" (P9 [158]). P9 similarly said that he would continue to use his iPhone, even if he found out that it was collecting a lot of data: "But I would buy it anyway, if there is no equivalent" (P9).

Furthermore, deciding to buy non-smart devices is equally becoming increasingly difficult, as for example a majority of TVs on the market already have smart features [97], and this trend is likely to continue: "As smart devices and appliances become more and more normalised, there is an increasing 'erosion of choice' for individuals who would have preferred their 'non-smart' versions" (Office of the Privacy Commissioner of Canada [135]). These examples show that there is a strong need for control over smart devices' communication behaviour.

3.3 Lack of Tools

While the previous sections have shown the importance and value of transparency and control, none of the tools known to the participants or studied in existing research provide the right kind of information in a usable way. This section discusses several issues that surfaced during the interviews.

3.3.1 Wrong Abstractions and Too Much Effort

The main problem is that current tools don't provide the right kind of abstraction and require too much time and knowledge from users in order to gain insight into their devices'

communication behaviour, placing the burden of protecting their privacy primarily on themselves [66], creating barriers of adoption: "At first you're still thinking about it, but after a while you just click on 'allow,' without giving it any thoughts at all. And then you just throw it away because it's pointless... because it just gets on your nerves [laughing]" (P5). Especially if such tools do not provide all the necessary information to make privacy-related decisions: "Because I'd be afraid that I'd have to constantly make decisions that I don't want to make, or just, yeah... it's just so... Uff!" (P7)

For example, while Wireshark, a low-level network packet analysis tool, would theoretically provide all necessary information for P5, he found it too time-consuming to get a bigger picture of what is going on in the network: "It's rather difficult to get some context. You see that it's sending something to all these addresses, but that doesn't tell you anything at first glance. [...] I had all the information but that didn't mean anything" (P5). Likewise, P4 and P2 argued that it requires concrete initial knowledge of what to search for in order to navigate through the large amount of data: "It's just insanely much information. You have to know what you're doing and know some queries to get some insight... that's rather difficult" (P4). "You really have to look hard to find what you need. It's rather difficult to get an overview" (P2). However, with Wireshark's slogan "Go Deep," such big picture is simply not its main goal [165].

These high requirements for time and knowledge were also mentioned by Lau et al. [87] and Tabassum et al. [150], where they argued that the required effort to use such tools outweighs users' concerns, especially since users have little awareness of the actual risks, as already mentioned in Section 3.1.2: "I can take a smart speaker and have a network firewall [check if the speaker is doing what it says it is doing], but that's additional work and expense and it demands an increased level of vigilance from me. I just don't see that as being worth the trade offs" (NU09 [87]). Complicating things even further, Tabassum et al. found that only participants with expert knowledge even knew of the available tools [150].

While Little Snitch (see Figure 3.1), a tool used by P1, requires less expert knowledge due to its usable interface, which, for example, groups data transfers by application and domain, P1 still complained that he had to manually check each domain for potential privacy risks or violations, which in his opinion, is again simply too time-consuming and would require him to almost become an expert himself:

> I mean, you have to consider that most people don't have the time to research on the Internet where the website originated from and what that means exactly. [...] Everybody has somehow better things to do than to start investigating in such detail that you almost have to become an expert yourself. (P1)

Glasswire (see Figure 3.2), a tool previously used by P7, similarly suffers from the same problem as Little Snitch, because P7 didn't know for which purpose certain domains were contacted, or if any could be considered unwanted. Since the tool treats all communication

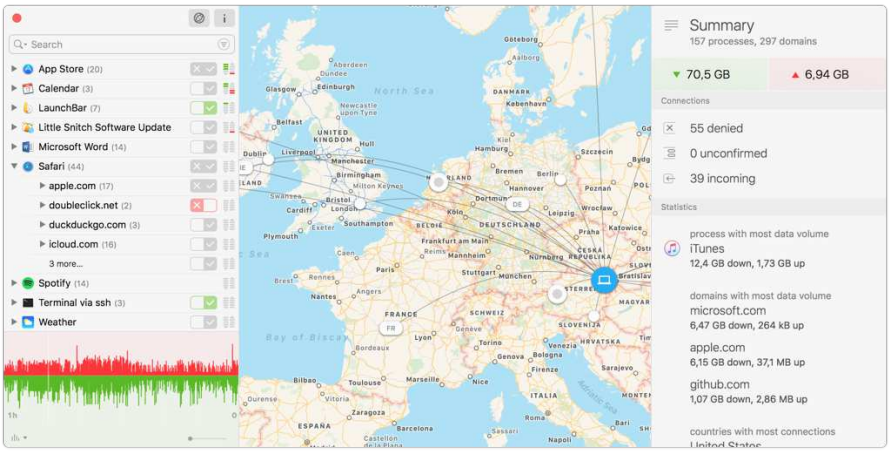


Figure 3.1: Little Snitch's network monitor [114], which groups contacted domains by application, but doesn't show any further privacy-related information about them, requiring users to investigate each domain on their own.

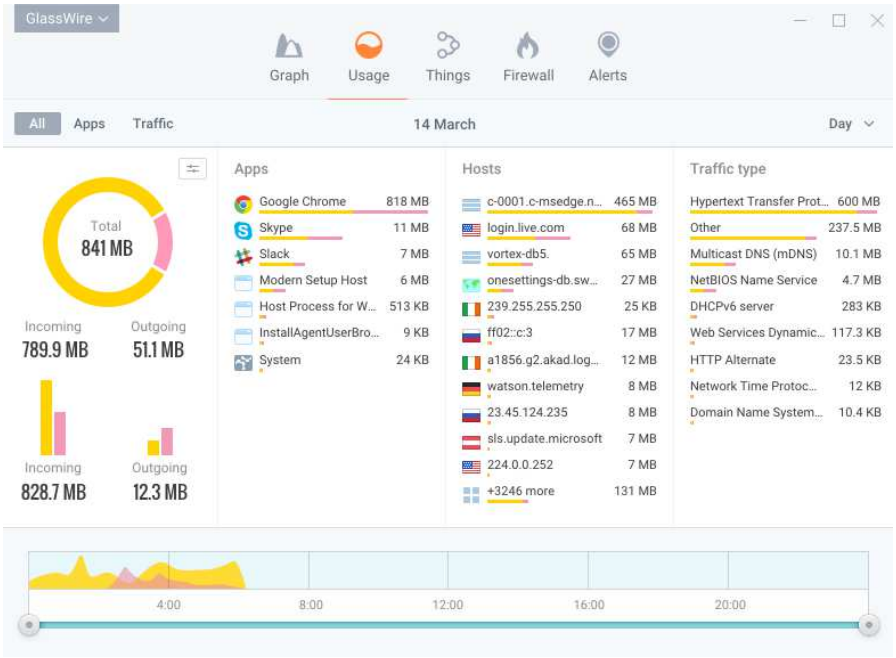


Figure 3.2: GlassWire's data usage view [55], which similarly treats all connections equally without highlighting potentially unwanted ones.

equally, P7 worried that unwanted one might be buried under the essential, more prevalent one:

> I think it probably would be even better to display *less* information, especially for something like this. Because, for example, "Dropbox is sending data again" – I already know that. Then perhaps truly interesting information somehow gets lost. [...] Exactly, less but more relevant one! (P7)

3.3.2 No Support for Smart Devices

Furthermore, none of the aforementioned tools can actually analyse traffic for the whole network, which makes them unsuitable for smart devices – a gap which has already been discovered in existing research [2, 66, 150]: "With my PC and phone, I have an anti-virus [installed], but I don't know how you could protect a speaker" (P1 [2]) "In the home environment you don't really have that much control over your privacy with IoT devices" (Florian Schaub [135]).

In order to support smart devices a tool would need to perform such analysis at a central location in the network, similar to a router, as some participants suggested: "For this you would need an appropriate tool, which is ideally integrated in... the router or something like that. I think that's probably the best point" (P2). P9 further added that such centrally located tool would also allow him to learn and configure just a single user interface rather than one for each device, saving him time and effort: "I believe you'd rather do this when you have your devices in a single tool, instead of having to manually configure each device separately what data... whether or not it's allowed to send data" (P9). Tabassum et al. [150] reported the same burden of having to manually configure all devices in a smart home, with Zheng et al. [172] reaching the same conclusion that such centralised management could be a first step towards a usable privacy control for smart homes.

While Wireshark, or any other packet capture software, could be run at such a central location, it would require a high degree of technical knowledge and effort, and still has all the aforementioned drawbacks. A more suitable tool is, for example, Pi-hole (see Figure 3.3), a DNS sinkhole, which can block advertising and other unwanted communication for all devices in the network, but still only provides domain names as abstractions, and moreover blocks communication primarily through existing blocklists [64], which is in conflict with participants' desire to individually block communication based on its context, as discussed in Section 4.5.5. The Upribox, which was used as a basis for the practical implementation (see Section 6.2), has similar drawbacks, as it silently blocks communication with no way for users to influence its behaviour [33], in addition to its lacking visualisation capabilities, as shown in Figure 3.4. And even current research projects such as the Princeton IoT Inspector (see Chapter 8) still don't meet participants' requirements for a usable abstraction [67]. Chapters 4 and 5 are meant to close this gap.

Time	Type	Domain	Client	Status	Reply	Action
2019-05-31 15:37:23	A	web.vortex.data.microsoft.com	192.168.0.23	Blocked (gravity)	- (0.3ms)	Whitelist
2019-05-31 15:37:18	A	mobile.pipe.aria.microsoft.com	192.168.0.23	Blocked (gravity)	- (0.4ms)	Whitelist
2019-05-31 15:37:18	A	wpad.home	192.168.0.23	OK (cached)	NXDOMAIN (0.2ms)	Blacklist
2019-05-31 15:37:16	A	client.wns.windows.com	192.168.0.23	OK (forwarded)	CNAME	Blacklist
2019-05-31 15:37:14	A	web.vortex.data.microsoft.com	192.168.0.23	Blocked (gravity)	- (0.3ms)	Whitelist
2019-05-31 15:37:13	A	app-measurement.com	192.168.0.160	Blocked (gravity)	- (0.3ms)	Whitelist
2019-05-31 15:37:04	A	web.vortex.data.microsoft.com	192.168.0.23	Blocked (gravity)	- (0.3ms)	Whitelist
2019-05-31 15:37:01	A	dxp86gw5pke1r.cloudfront.net	192.168.0.160	OK (forwarded)	IP (33.8ms)	Blacklist

Figure 3.3: Pi-hole's query log likewise only shows domains without additional information, forcing users to investigate on their own if they might be unwanted.

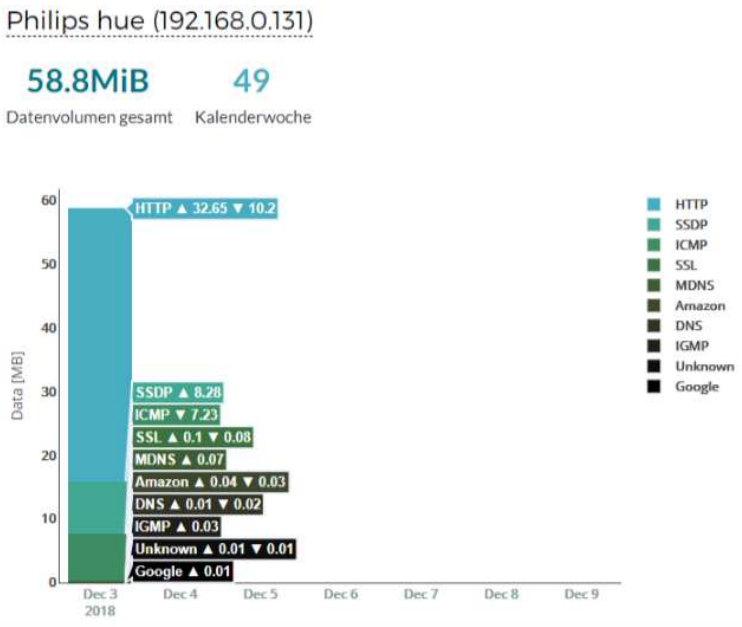


Figure 3.4: Upribox's device details, providing only basic classification of a devices' communication behaviour, with the majority just shown as "HTTP."

CHAPTER 4

Identifying and Controlling Potentially Unwanted Device Behaviour

The previous chapter strengthened the need for transparency and control over devices' communication behaviour, and showed that existing tools don't provide the right kind of abstractions for participants to make privacy-related decisions on their own. This chapter uses the theory of Contextual Integrity (CI) to create a data abstraction that allows users to identify potentially unwanted device behaviour by describing the appropriateness of information flows in specific contexts through five interrelated parameters: the *data subject*, the *sender*, the *recipient*, the *transmission principle governing the transfer*, and the *type of information sent* [6, 98, 112]. In this thesis about smart home IoT devices, similar to Apthorpe et al. [5], the data subject is fixed to the owner of the device and the sender to the device itself, as participants mentioned that data leaving their devices is more relevant than the data they receive. Using the remaining parameters, the following sections describe in detail how participants identified potentially unwanted communication behaviour, how they imagined a usable control, and what concerns they have about the trustworthiness of such a tool. The next chapter presents how participants visualised such usable transparency and control using these abstractions.

4.1 Recipient

Starting with the recipient of information, some participants naturally wanted to know the contacted organisation, and others like P12 concluded themselves after some discussion that organisations would provide a usable abstraction of smart devices' communication behaviour: "Yeah, okay. You're right. You convinced me now. [...] Especially Facebook and Google. I *don't* want my Fire TV Stick to communicate with Facebook" (P12). As

Van Kleek et al. [159] already reported, organisations allow participants to reuse their existing mental models and preferences about brands and ecosystems. P10, for example, mentioned that, while he thinks it's unreasonable, he prefers to share his data primarily with Google, since they already know everything about him, arguing that there would be no harm in sharing even more. Existing research found similar data sharing choices based on personal preferences and previous interactions with organisations [87, 150, 158].

> Google, for example, already knows everything about me. It doesn't matter a damn if they knew even more. But Facebook... hmmmm... I figure – rather a little reduced, not too much. But Google... okay, someone knows everything anyway, then at least it's all in one hand. (P10)

In addition to personal preferences and in line with CI, P12 emphasised that his acceptance of the recipient also depends on the concrete device, with him, for example, disliking if his Amazon Fire TV stick would communicate with Facebook, while not generally distrusting the organisation itself: "I see *no* reason why it should communicate with Facebook. It's the combination. Because it's not that I distrust Zuckerberg... but the combination just doesn't work" (P12).

Likewise, P10 described how he generally dislikes data being sent to organisations that are not directly related to the services actively used by his devices: "But the data has no place on some other server that is not obviously related. [...] When I don't use Apple Music or Spotify, the device should not interact with them" (P10). This distrust to unrelated third parties has been similarly identified by existing research [38, 87], and is further discussed in Section 4.3: "When they start giving it to third party users, and I don't know exactly who they are or what they're doing, that's probably where I start to get concerned" (U07 [87]).

Finally, for unfamiliar organisations some participants pointed out that the name alone is not sufficient to support privacy-related decisions, which was equally reported by Van Kleek et al. [159]: "If [the domain] is registered to 'Hans Maier,' then I'm not sure how useful that is for me. Or which organisation it's registered to – I don't know if you could do anything with that" (P7). Likewise, participants in the study of Schaub et al. were confused about the large number of unknown organisations shown by privacy-focused browser extensions [140]. The following sections therefore describe three attributes mentioned by participants that would influence the trustworthiness of an unknown organisation: its *country of origin*, its *prominence*, and its *subsidiary relationships*.

4.1.1 Country of Origin

To judge the trustworthiness of unknown organisations, some participants found their countries of origin helpful, and had second thoughts, for example, about Far East organisations: "There are indeed some regions where you might think twice – okay, what on earth is it doing in Bangladesh" (P10). "Especially for smaller devices like a robot vacuum. It would be weird if it sent audio recording to a server in China" (P7).

> If I get an app, I'm always careful to see if the developers are from mainland China... They seem like independent companies, but all the people know that these companies are controlled by the Chinese government – and they have very large centres for consolidating people's data. (P18 [158])

In the same way, some participants also worried about weaker data protection standards in certain countries compared to the European GDPR, which could make unwanted data sharing easier and more likely, a concern similarly observed by Van Kleek et al. [158]: "Because we do have data protection standards in Europe and maybe America... compared to some places in Africa or somewhere. [...] That simply because I registered with this provider, they may now sell my health records, just because it's allowed in that country" (P9).

However, judging an organisation just by their country of origin might not only be unfair, but might also excuse otherwise misbehaving organisations, as P10 noted:

> That doesn't mean that all Russians are totally bad or all Americans or Europeans are totally awesome, just because they might have their company headquarters there. That basically means nothing at all. [...] That's extremely difficult.. And I believe, even unfair towards the organisations. (P10)

4.1.2 Organisation's Prominence

An organisation's size or prominence is seen as another useful attribute for judging unknown organisations. P6, for example, mentioned that already established organisations would probably invest more time and money into securing their products, since they have a reputation to lose in case of data leaks or hacks, putting, however, smaller organisations with less budget in a more challenging situation, since Rosner [134] argued that "the preservation of privacy can never be divorced from economic considerations."

> In my opinion, larger companies like Apple, Samsung, will meet certain security standards better than other competitors. The smaller the organisation or the lower its market share, the greater the risk, because they don't have that much budget or maybe don't want to invest it to secure their products, to really ensure nothing happens. So that their reputation isn't ruined. (P6)

Likewise participants in related studies reported similar concerns about lesser known organisations [150, 158, 172]: "There's some comfort in Google being a brand that you recognise, as opposed to [company name]. Like, who are they?" (P9 [159]) "For the *big* companies, I'm sure the regulation is more strict and a lot of people are really carefully watching them for what they are doing, so I feel they're probably relatively safer than whatever or whoever these are [right]" (P15 [158]).

4.1.3 Subsidiary Relationships

Furthermore, Van Kleek et al. [158, 159] identified that an organisation's subsidiary relationship can also influence users' confidence in unknown organisations, if they already trust one of its parents: "It was useful knowing it was [owned by] Twitter" (P10 [159]). "This stuff, I mean... It's kind of interesting these sites – like *Bitstadium*? Oh, that's *Microsoft*!" (P21 [158]) Likewise, Binns et al. [13] argued that such information might also negatively influence users' perception of an organisation, if they are, for example, trying to avoid data being "sucked into the all encompassing data mountain of Google and attached to my very persistent Google identity," as P18 [13] put it. And because results from Razaghpanah et al. [128] showed that it's common practice for subsidiaries of an organisation to share data with each other, such subsidiary information become even more relevant.

4.2 Transmission Principle

In addition to the recipient, the transmission principle, which defines the condition governing the transfer of information between sender and recipient, is another influential CI parameter to identify potentially unwanted communication behaviour. Examples include "if the data is kept anonymous," "if it is used for product improvements," or "if the subject has given consent" [5]. This section focuses on the last two examples: the purpose and consent of data collection.

4.2.1 Purpose of Data Collection

The purpose of data collection and transmission was reported as an important determinant of acceptance not only by most participants but also by a variety of existing research [79, 97, 169]: "I would prefer – 'it transfers this' and the *purpose*. I think that would mean a lot more to me, and it would increase my trust in the device" (P7).

Some participants were especially sceptical about their own benefit from the data collection, and, in line with existing research, would only accept such collection if they get appropriate value in return [97, 150, 172]. P11, for example, at least expected his device to be improved or adapted to his habits in return: "Then I expect that they're not just collecting the data to again make some money off me, but that it's improving the device. That it's adapting to my usage patterns. That it's helping me, and not that they're reselling me as a product" (P11).

However, recent news reports and public outcry over contractors reviewing voice recordings of IPAs [28, 61, 157] have shown that using data for the seemingly beneficial purpose of improving services and device functionality does not automatically make users accept such practice, if it violates their expectations regarding data processing; in these examples, other people listening to their private interactions and conversations, a practice similarly reported as unexpected by Malking et al. [98]. Amazon even allowed workers to review footage recorded by their Ring smart security cameras, with the report suggesting that

workers might even have had access to live feeds [12]. At the time of writing, the public outcry made Google [141], Apple [16] and Facebook [49] to temporarily halt the manual reviews, while Amazon [26] added a opt-out option and improved the wording of their disclosure.

Consequently, data collection only for the benefit of the recipient is largely considered negative: "Whether there is a reason, so to speak, so that it's necessary for the program to function, or if it's just a data collection obsession, which actually only serves the organisation itself" (P1). With targeted advertising seen as a prime example of data collection with little to no benefit for users [5, 97, 150], as further discussed in Section 4.3: "If it says it will be sold for advertising purposes to other organisations, then I'd really think twice about that" (P11).

> They have put a lot of money in this product, and then they are selling it. So, they must be using it for something other than me telling my house to turn on my bedroom light. They are building advertising model of me. They want to know who I am and how I work so they can try to sell me something. (Id19 [150])

Yet, some participants in the study of Zheng et al. [172] saw a potential benefit in such targeted advertising, if they, for example, get better deals in return: "It feels as a consumer, if there is a better, more seamless experience to advertise to me, like I should benefit from that in some way. Or there's that relationship where if I can get something out of that, then that seems like a better deal" (P8 [172]). Likewise, Lau et al. [87] noted that some participants found targeted ads useful and preferable to irrelevant ads, which again demonstrates that privacy preferences are highly individual.

4.2.2 Data Collection Without User Interaction / Consent

In addition to the purpose of the data collection, some participants were further interested *when* their devices communicated, generally distrusting automatic data collection, i.e., collection not caused by interacting with their devices: "If the device now suddenly starts to communicate with the Internet without using the remote control, then that's a different story. [...] It *mustn't* do anything if we don't use it" (P12). Page et al. [117] supported these concerns by showing that some people still see their smart devices as tools instead of autonomous agents, not fully trusting them to make decisions in their best interests. Likewise, Apthorpe et al. [5] found that user consent is still a key factor in the acceptance of data flows occurring in a smart home. This distrust influenced some participants to create a time-based visualisation, which is discussed in detail in Section 5.5.

4.3 Third Parties and Tracking

Even though the previous sections have shown that most participants dislike data transmission to unrelated third parties as well as data collection with no apparent benefits, such practices are very common in the app and web ecosystem [80, 101, 128], and recent news reports have revealed that smart TV manufacturers already calculate "post-purchase monetization" into their price model, i.e., they mine users' behaviour to allow advertisers to target them based on a variety of properties [119]. And there are reasons to believe that such monetization practice will be applied more widely to the future generations of smart devices [66, 121], especially because these devices are able to collect personal information in real-time, directly from inside consumers' homes, which could enable more advanced tracking features like detecting emotions, which has been dubbed the "Holy Grail of marketing" [135], and is already being explored by big tech companies like Amazon and Google [27, 94].

Moreover, such tracking behaviour is not only highly opaque to users but to developers as well, leaving them often unaware of data collected by third-party libraries [5, 118]. A report by Privacy International, for example, demonstrated that the Facebook SDK[1] may unknowingly collect personally identifiable information about app usage, and share it with Facebook, whether users are logged into Facebook or have an account at all [75]. A related investigation by a German consumer advocacy group also criticised that the opt-out mechanism for the SDK only sets a flag to prevent the usage of the data for targeted advertising, without, however, stopping the collection itself [137].

P6 additionally worried that by involving third-parties in the data collection and storage, as opposed to just directly sending the data to the manufacturer or developer, the risk for data leaks or misuse increases, as similarly reported by Rosner [134]:

> For example, if Sony hires another organisation: gather the data for me; then I'm less secure as if I'm directly sending it to the manufacturer. As if I'm sending it to third or fourth parties. Because every interface, every storage somewhere, raises a certain scepticism. (P6)

Razaghpanah et al. [128] further showed that much of the information collected by trackers ends up in the hands of data brokers, which aggregate and sell information about users for targeting purposes, and may even additionally combine the data with records of offline purchases [46, 127]. Such aggregation may allow unrelated organisations to gain detailed insight into potentially private information about users, as P1 mentioned:

> [Concerning Facebook as a tracker] This can sometimes reveal things that are very personal and have no place at Facebook; e.g. information about medications that can be used to draw conclusion about which illness one has.

[1] https://analytics.facebook.com/

> This may result in a very clear picture about a person. These are some very private details that have no place at an unrelated organisation. (P1)

Consequently, P10 worried that such aggregated data may ultimately be misused by insurance companies, for example, who could use the data to enforce worse conditions, which Peyton [122] similarly mentioned as a threat. Some insurance companies already go so far as to include fitness trackers directly into their plans [9].

> That's the thing. It can be misused. And then, worst-case scenario, it winds up at insurance companies and then they say – if you do this and that, then you get these and those conditions... And once someone starts, others will follow. And this can be misused in so many ways. (P10)

However, providing transparency of third parties' tracking behaviour is seen as a major research challenge, because it not only involves a multitude of interrelated organisations that exchange data with each other in an opaque manner [20] (see Figure 4.1), but research has also shown that users have difficulty understanding the different purposes of such organisations like "retargeting" or "analytics," suggesting the need for better visualisations and explanations [140].

4.4 Data Type

Finally, the type of data, the last remaining CI parameter, was also considered by most participants as a deciding factor in their acceptance of a data collection: "For example the TV, if it sends pictures of the webcam to some place and how often. [...] Yeah, ideally, I would like to know what kind of data" (P2). And in line with existing research, a majority of participants saw audio and video as particularly sensitive, worrying, for example, about always-listening devices, as already mentioned in Section 3.1.1 [87, 98, 150]: "However, if it concerns, for example, microphones, then I would have *very* strong reservations. I would, for example, never put some Alexa in my home. I mean, with that you even know that it sends every spoken word somewhere. That's another level" (P5).

Other participants were similarly worried, for example, about smart cars tracking their location, smart fridges sharing their eating habits, or biometric information falling into the wrong hands: "Even with a car, for example, I wouldn't want those to be so interconnected [Concerning GPS] Yes, exactly. I wouldn't see the value to risk that" (P8). "What I absolutely don't want are any options on (smart) fridges, where someone automatically checks what I have stored" (P6).

As with the other CI parameters, the type of data is equally influenced by the others, and participants felt comfortable, for example, sharing audio recordings of their devices when they serve the purpose of the device, such as controlling it per voice commands: "It makes sense if I own a smart speaker that it transfers audio recordings. [...] It makes sense that it's sent to Amazon for it to work at all" (P10).

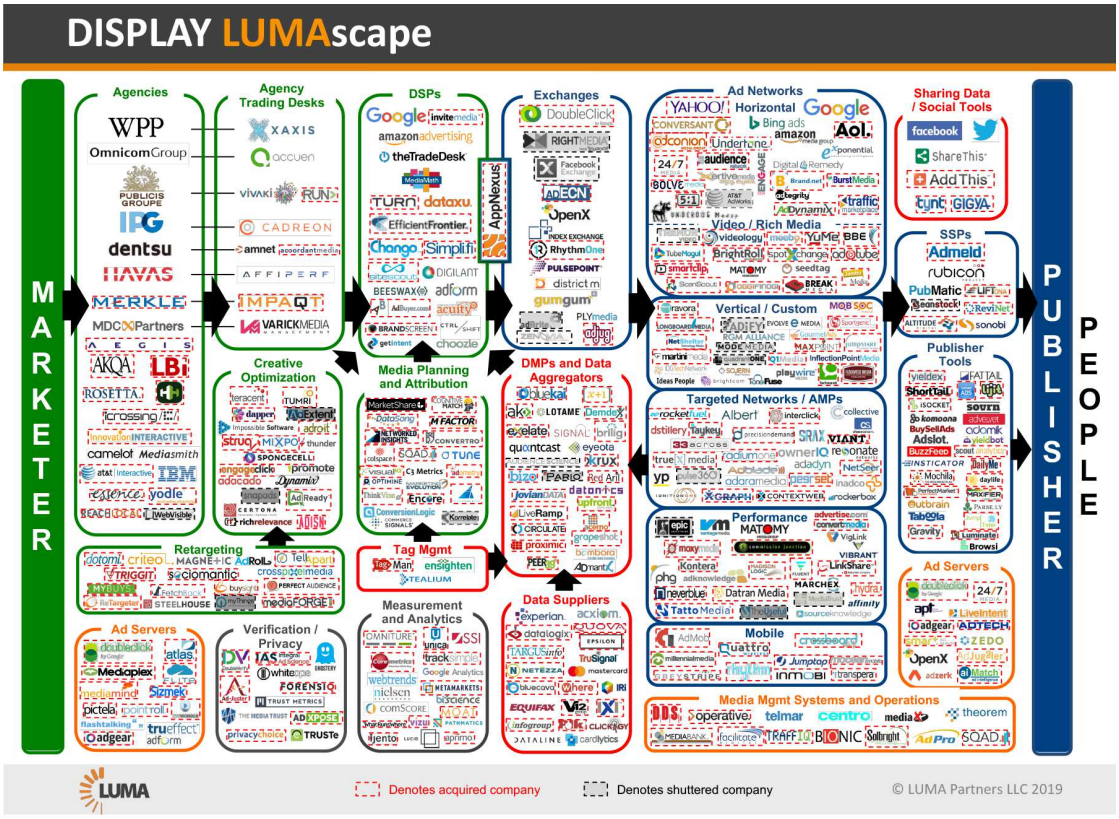


Figure 4.1: Display advertising landscape showing the complexity of the tracking ecosystem [93]

However, identifying the transferred data type is a major research challenge, because related research about detecting sensitive data leaving smartphones has already shown that it requires access to the decrypted data stream [118, 131, 149], which is not easily accessible for smart home IoT devices, as in most cases they don't allow direct access to their operating systems or the execution of third-party applications [8, 168]. The remainder of this thesis therefore primarily focuses on the recipient and the purpose of data collection. While this neglects important aspects of CI, Apthorpe et al. [5, 6] showed that most participants see advertising-related data collection generally as unacceptable, regardless of the transferred data type, which inspired the highlights in the tool of this thesis (see Chapter 7).

4.5 Usable Control

In addition to usable transparency, participants also desired to assert control over such unwanted communication behaviour, but not every form of control is desirable. This section therefore describes participants' requirements for such usable control.

4.5.1 Retaining Smart Functionality

Most importantly, such control would need to work at a granular level, blocking only unwanted functionality like tracking, while still being able to use the device's smart features, as P8 stressed:

> So when we buy a new TV, and I could choose it's connected to the Internet for Netflix, YouTube,... *But* it doesn't send any data to any organisation that would reveal my viewing habits. I'd prefer that. But I have no idea how to do that. (P8)

Ghiglieri et al. [53] similarly showed that participants were not willing to forego all smart features just to have their privacy protected, but they would invest time and money for privacy protections that retain their smart devices' Internet functionality. P1 likewise said: "That's of course not what users want. It would very well be possible to provide the Internet and still don't collect any data" (P1).

4.5.2 Coarser Granularity

A technical suggestion to minimise the required technical knowledge and time is to offer blocking on a larger scale such as blocking whole organisations without having to consider every associated network connection: "Here it would be nice if you didn't have to click on each one individually, but if you know all these servers have the same purpose, more or less, and belong to this organisation. And then just say, block all servers of this organisation" (P1). This approach would provide a higher abstraction above an IP-based firewall and was similarly suggested by Seymour [144]: "Yeah, that'd be a nice feature, because then I won't have to do that anymore. And try explaining to people how to configure a firewall" (P12). "Perhaps I can use a menu item to configure that my devices generally don't send any data to Facebook or Google etc" (P9).

Besides organisations, some participants also wanted to block the transfer of certain data types, such as audio recordings, an approach similar to the smartphone-inspired permission system envisioned by Zheng et al. [172], restricting devices' sensor usage as well as who can receive such data: "This means that I don't want microphone data to be transferred from apps, for example. I don't want them to transfer image data, video data, or something like that" (P10). "If some traffic is classified as microphone data, then I want to block it. But if some traffic is classified as 'change of channel,' then I don't care and it can pass" (P5).

4.5.3 Blocking All Communication

P9, P11, and P12 went even further and proposed mechanisms to temporarily block any communication of a device in certain situations, comparable to an "emergency stop" button, as P12 put it:

> In a panic, you either rip off the power, or unplug the network cable – okay, that's not gonna work for WLAN. So if I really panic as a user, I would like to have such an emergency stop button. A global one, for the entire network. [...] [Or] maybe just for this group of IoT devices. (P12)

This would ensure that no more data will be transmitted, even if it means that devices stop working:

> For example, if I go on vacation. If I no longer want the robot vacuum to be online... then I cut the connection here. And I don't want it to send anything anywhere anymore for sure. [Question about functionality] Yeah, no, then the devices simply don't work anymore... cutting it completely, yes... then you're sure nothing will happen anymore. (P11)

This desire to disconnect smart devices from the Internet was also described by Yao et al. [169], where some participants mentioned that it would give them "peace of mind," knowing no data can be transferred. Similarly, some participants in the study of Lau et al. [87] resorted to unplugging their devices to really make sure they stop recording: "If I really wanted to, I'd unplug it. You can hit the mute button on top. I don't know if it's actually doing anything or not" (U15 [87]).

Based on his idea, P12 also envisioned a "holiday" or "away" mode, which would prevent devices from sending data on their own, while he is still able to, for example, check the surveillance camera or set the thermostat remotely; basically blocking only automatic outgoing background communication (see Section 4.2.2).

4.5.4 Official Mechanisms and Regulatory Approaches

To overcome the burden of using Privacy-Enhancing Technologies (PET), some participants also suggested that devices should directly give users official options to disable unwanted or unnecessary communication behaviour, which would further avoid any potential issues or side effects from manually blocking communication: "For example, if an app has some optional communication behaviour, that I can simply turn it off without any manual intervention, as otherwise it might cause unwanted side effects" (P2). "Because I would also partially block functionality with that; then I maybe won't be able to do everything, and it might get error-prone" (P7).

However, P3 interjected that such official options might be in conflict with manufacturers' interests, as the collected data might be of value for various purposes: "Yeah, but I mean, why would the organisation building the TV set or whatever even include such functionality in the first place?" (P3) "Ninety percent of the time, such information is probably worth something. For the market and media; for advertising experts and analyses" (P6).

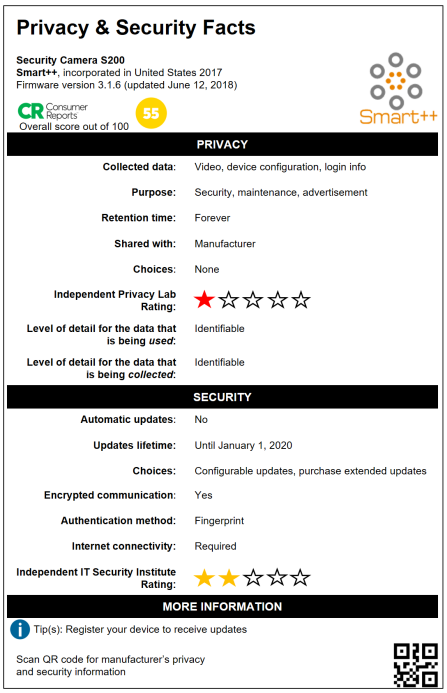


Figure 4.2: A proposed privacy label for a smart device with poor privacy and security practices [39] to improve transparency about its communication behaviour at the time of purchase.

To solve this conflict of interest, P1 suggested to generally ban invasive data collection by law, lifting the burden of protection from users' shoulders, similarly suggested by some participants in the study of Yao et al. [169], which would also prevent manufacturers to just bury any official options in complicated menus, betting that most users don't invest the time to find and change them.

A less restrictive regulatory approach are privacy labels (see Figure 4.2) that try to summarise relevant privacy and security properties of a device in order to improve transparency about a device's communication behaviour at the time of purchase [39]. P9 and P12 similarly imagined privacy seals that would verify certain privacy-related properties of a device, so that users don't have to manually research such information and decide if it meets some set of guidelines. Projects like "The Digital Standard" [151] and Mozilla's "*privacy not included" [108] already attempt to create such guidelines for testing smart products in order to help users make more informed purchasing decisions.

> As an end user I can't read and check everything anyway, but I would like to have some kind of logo on it, like the AMA seal of approval for meat, indicating that all information has been disclosed by this device or software [...] and other people have already looked at it; it's well-known and only few people have complained. (P12)

However, P12 noted that frequent and automatic device updates would require a re-validation of such seals, since it has already been shown that software updates may introduce new and unwanted device behaviour [42]. And despite their use, such external assessments can't fully consider users' individual privacy preferences, making general assertions about privacy properties difficult and usable tools for end users an important contribution.

4.5.5 Manual Decisions but With Recommendations

Considering the ideas of all participants for controlling devices' communication behaviour, almost all of them disliked automatic mechanisms and instead preferred to decide for themselves what communication to block, based on the provided information and the specific device: "I must be able to make my own decisions. This means that it must be up to me whether or not I block it. And not someone else should decide that for me" (P10). "This means that the app itself does not intervene automatically, but I can click on a server and look at its details, and then based on that I can say – yes or no, I want that or not" (P7). P6 similarly argued that he might value certain optional communication, if it, for example, enables the manufacturer to diagnose and fix problems with his device: "Because maybe I don't want to block all of that. Maybe I would like to use some of the tools... or some of the conveniences. [...] Not everything has to be bad... Manufacturer, update, and device information are also important" (P6).

But while the tool should not automatically block any communication, most participants would like to get recommendations and warnings about potentially unwanted behaviour: "Maybe even with a recommendation – this server collects this and that, or maybe there are some concerns about this server, you should block it for security reasons" (P9). "Yeah, details and maybe also tips on whether this is some known malware or if it's already somewhere in a blacklist or so. [...] Attention, the connection should be checked or some actions should be taken" (P6).

However, such recommendations should also be ignorable, if the communication is deemed acceptable, as P3 argued: "That you can then say – Okay, no, I don't trust that anymore. Or I always trust it, because I know it's OK, even though it says it's not" (P3). Details on how participants imagined such recommendations are discussed in detail in Section 5.2.

4.6 Trust in the Tool

Finally, all except three participants (P1, P6, P12) voiced some concerns about trusting a tool that would provide such transparency and control, worrying not only about how it processes the data but also if the provided information is accurate.

4.6.1 Trust in Data Handling

First of all, a concern is that such tool could itself create additional privacy issues, if it doesn't handle the collected data with care, or even uses it for other purposes

than expected. Murman et al. [111] similarly questioned why users should trust a tool providing information about devices they don't trust, arguing that it would complicate the "chain of trust," similarly mentioned by P3: "But then someone must again tell me that this tool is trustworthy. I think that's a circular argument" (P3). Likewise, some participants of Schaub et al. [140] and Malkin et al. [98] voiced similar concerns about the trustworthiness of a tested privacy extension, arguing that without the extension they might not have the same protection, but at least they don't have to worry about giving the tool access to their private data:

> I don't think I'm concerned about the New York Times. I'm concerned about Disconnect having this information all in one place about all of the websites [...] If the plugin is not installed, they're not stopping anything, but they're [the extension itself] not housing anything either. (P6 [140])

To improve users' trust in a tool's data handling, Chetty et al. [19] and Yao et al. [169] recommended that it should process and store data only locally, avoiding potential security and privacy risks of transferring data to a remote location. Failing to do so, caused participants of Chetty et al. [19] some anxieties and concerns: "Having all your data logged on a server, your whole usage patterns, and everything it's a bit nerve-wracking" (H8 [19]). "Who has access to it? And what is the control, my control over that data? And what are they using the data for?" (H6 [19]) A current negative example for data locality is the Princeton IoT Inspector research project (see Chapter 8), which transmits its collected data to their server [67, 125], resulting in some of the following commentary in a related news report: "Who will be spying on the anti-spying software?", "While it is spying on you?", or "And exactly what other things does this app track that they don't tell you about?" [126]

4.6.2 Trust in the Provided Information

Furthermore, some participants were also sceptical if such a tool would provide accurate information: "[Showing the organisation for an unknown domain] So how do *you* get that information? [...] How much can I trust the tool that this is really correct?" (P8) P3 further added that there might be a conflict of interest if the tool is created by the same company as the device it should analyse:

> P2: I would then like two of these devices so that one device can monitor the other [laughing]
>
> P3: But you'd actually have to get one from another company, right...?

To increase trust in such tool, P8 suggested that it would need to be created by a trusted third-party like "Stiftung Warentest," an idea also supported by existing research that recommended independent research labs or trusted entities like "Consumer Reports" as the maker of such tool [39, 159, 169].

A different approach for increased trust was proposed by P5, who suggested that such tool should provide insight into the steps it took to infer the displayed information:

> As long as I can rely on the software that it correctly classifies it as Google. [...] The question is – can I check the software or do I have to trust it? Because if it says, this is the IP address, it maps to this domain name, which belongs to this company...[Basically an explanation] how the software concluded this belongs to Google. (P5)

As an example of a breach of trust, P7 half-jokingly suggested that such tool could use a business model similar to that of the *Acceptable Ads Exchange*, which takes money from advertisers to have their ads excluded from blocklists [1]:

> [Not entirely serious] If you write an app like that, I think you can make some good money, because it could use a similar business model as these ad blockers – the vacuum cleaner manufacturer, for example, gives you money and then you hide some servers... something like...[Acceptable Ads]. (P7)

These concerns show that such tool not only has to collect and analyse its data in a privacy-friendly way but also provide its results in a way that's accountable to the user. Making the source code of such tool open to the public could be a first step in the right direction. How the tool of this thesis further tries to increase its accountability is discussed in detail in Chapter 7.

CHAPTER 5

Participants' Visual Encodings and Interactions

The previous chapter laid out what participants regarded as unwanted communication behaviour and what information they would need to feel informed. This chapter discusses how participants imagined visualising such information by presenting the sketches they created during the participatory design exercise, grouping them in the following sections by their main theme.

5.1 Simple and Quick Overview

A key theme mentioned by most participants was a simple overview that quickly provides insight into a device's communication behaviour without having to dig deeper into details, as P10 mentioned: "So I don't have to somehow dig into the system, and have to investigate every app or every megabyte, why it's doing this or that... But it must be collected somewhere, so that it's immediately clear at first glance who is doing what" (P10). He further argued that people would not be interested in a visualisation, if it would require more than three clicks to figure out what is going on. P6 similarly argued that it's difficult to get an overview from an overly cluttered visualisation: "I certainly don't like overloaded presentations. I prefer simple and straightforward designs. Otherwise, the overview is quickly lost" (P6).

To realise such a quick overview, P9 suggested to display only the essential information up front, providing further details only on demand: "I think it's important to get a good overview quickly. Which is kept rather simple. [...] And then perhaps, starting from the devices, if you're interested, a deeper layer which shows details on *who* received the data" (P9). Shneiderman called this the *Visual Information Seeking Mantra*, which progressively shows more relevant and detailed information: "Overview first, zoom and filter, then details-on-demand." [145]

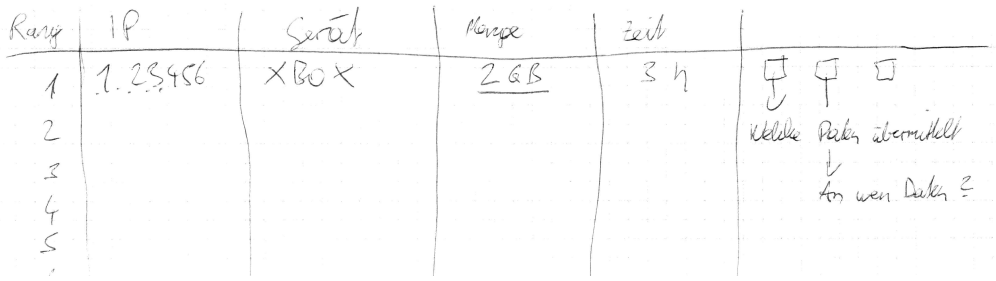


Figure 5.1: The overview part of P9's sketch, showing a sorted device list with basic attributes, providing further details on demand.

The overview of P9's design can be seen in Figure 5.1, containing a minimal table, which acts as a rank list for his devices, sorted, for example, by the amount of transferred data, additionally showing the duration of communication, the device name, and three buttons for further details. P3 similarly argued that details should only be displayed on demand, so as not to clutter the overview: "But I think that's some information you don't have to show to everyone, better hide it behind one of those 'show more' buttons [...] Because I believe that more information is sometimes even hindering" (P3).

One example for such details on demand in P9's sketch is a stock chart-inspired time series chart that opens when clicking on the amount of data underlined in his design, showing the distribution of the transferred data over time. Such visualisation would, for example, allow him to detect potentially unwanted communication patterns of his devices (see Section 4.2.2):

> This could be quite helpful, to get an overview of the day, maybe even with averages over a year over the time of day. So you can perhaps see that this is some data that is periodically sent at midnight, where I'm certain I haven't done anything with the device, as I'm normally sleeping. (P9)

To get further details, the first button on the right-hand side of his sketch opens a log that shows what type of data a device transferred and for which purpose: "I think that's rather futuristic, but maybe also that you can use different buttons to view a log of what kind of data has been transferred. [...] And for which purpose" (P9). The second button in his sketch opens a visually similar details view (see Figure 5.2) containing a device's contacted server and their corresponding organisations, again ranked, for example, by the amount of transferred data. He was, however, aware such view could potentially contain a lot of entries, and therefore imagined a filter that would only display important, unusual, or potentially unwanted communication, based on expected device behaviour and existing blacklists:

> That you get a list of communications that are, let's say, important. That are not common – perhaps some data that hasn't been transferred in the last

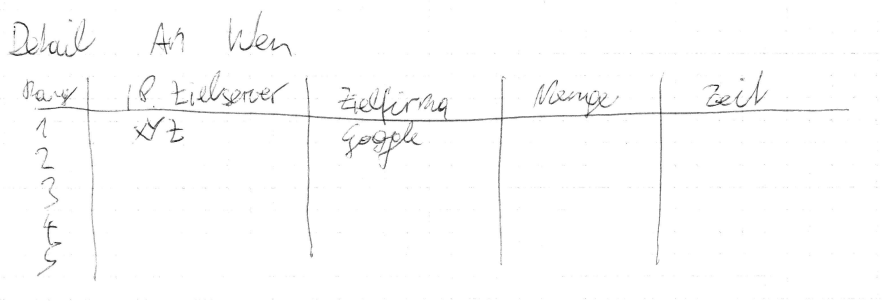


Figure 5.2: The details part of P9's sketch, showing contacted servers and their organisations, where he also imagined a filter to show only unusual or potentially unwanted ones to avoid cluttering the view.

> two, three weeks, or in the last year. So you could perhaps filter to only show IP addresses that are unusual. [...] Or servers with incoming or outgoing data that is unusual in your context, or for the device you have. (P9)

To further aid the discovery of relevant information, he thought of an alarm or notification that would inform him about the availability of such analysis results: "That as soon as this IP address appears or is logged in this program, that you get some information – because of course, from the second layer on there will be lots and lots of data" (P9).

P10 similarly realised a simple overview by using the concept of traffic lights (see Figure 5.3), which quickly signal via icons and colours if some further action or investigation is required by the user: "The main point for me was that it's visible at first sight, if there is something wrong. [...] Visually nice with red, orange, green. Very intuitive, because you know that red is bad, or in the sense that something went wrong. And green is okay" (P10). And in the case of red, his visualisation would also provide ways to dig deeper, to uncover its cause.

To detect such unwanted behaviour, each device has its own user-defined rules that define what is considered expected or unexpected; for example, a smart heating system, the third traffic light in his sketch, should, in his opinion, only communication with the teleheating provider Wien Energie, who provides the main functionality of the device, but not with any unrelated, external organisations:

> Considering my heating system, I probably only want it to communicate with Wien Energy. Because that's where it's controlled, and the data shouldn't go anywhere else. [...] If it's going to any external organisation, then it can already become red, because it has no business there. Because you only need connections to make it warmer. And not that someone else knows. (P10)

To simplify the configuration of such traffic lights, he imagined default rules for acceptable behaviour of each device, comparable to Manufacturer Usage Descriptions (MUDs) [89]:

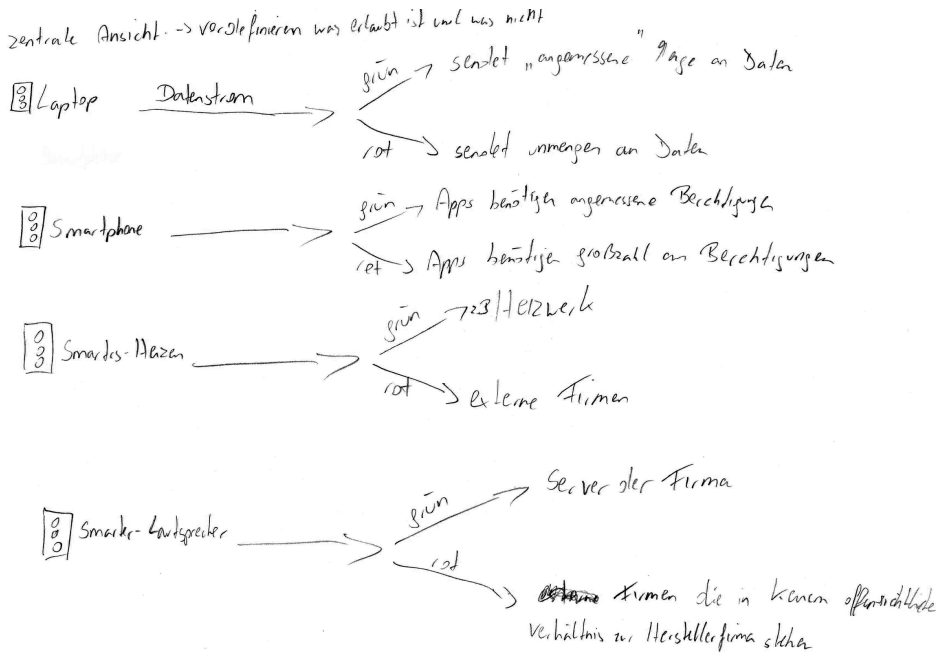


Figure 5.3: P10's sketch, showing a traffic light-based visualisation, which is meant to give a quick overview whether something is wrong with his devices, with further details provided on demand. The arrows on the right-hand side just visualise what happens in the background, and are not part of the design.

"It can already be predefined what is possible with the smart speaker, so that I don't have to set it up myself – okay, this is what you're supposed to have access to, this, this, this... because that's tedious," while, however, still retaining control over the actual enforcement in the end, "but I want to be in control so that I can turn it off. If I don't have Apple Music or Spotify, then it shouldn't access that, in my opinion." While his sketch also contains non-smart devices, like a laptop and a smartphone, he was, however, aware it would be rather difficult to define generally applicable rules for them, in comparison to special purpose smart home IoT devices:

> Of course that's more difficult with a laptop than with speakers. There you have all sorts of things. But coming back to the Internet of Things, then that's actually doable. Because I'm not buying such a device for a general purpose, but rather for some specific functionality. (P10)

5.2 Highlighting Potentially Unwanted Information

To provide such a simple overview, as identified in the previous section, most participants included design elements to signal potentially unwanted device behaviour, as manually

having to judge such information is seen as too time-consuming (see Section 3.3.1):

> Well, I think considering the amount of data alone, that's completely irrelevant. Who is gonna look at that? And I think there lots of households with at least the amount of smart devices we have. And you gonna lose any interest if you have to manually check all this information. [...] So, I believe that a manual analysis by the user isn't going to work at all. (P12)

> I think you'd have to make a lot of decisions for the user, whether that's good or bad, because I don't think... even if you have all the information – you see server XY from Google, or let's say, something more fishy... China server XY. Then the user doesn't know..., then even *I* as a technician don't know or can't really judge that. (P7)

In order to effectively communicate such potentially unwanted information, P3 stressed the importance of a usable visual encoding, like the colour red, icons, or explicit warnings – concepts already known from the information visualisation field [111] – or otherwise, she might not consider such information important enough to act on them:

> But it has to be broken down in such a way that *I* as an end user can understand whether that's bad or good, but I would... I mean, I wouldn't get that [...] Even if it's just written there, I'm not sure if people would pay enough attention to it. If there is no warning like "Hey, maybe watch out for that!" or with red, or I don't know... Something indicating it's bad so to say. (P3)

Figure 5.4 shows how P3 imagined such highlighting, which includes warning icons and the colour red (sketched using striped backgrounds) in both the device list on the left-hand, and the details view on the right-hand side: "Then I can quickly see here that something isn't right. [...] That it's not trustworthy, which is why there is this warning triangle" (P3). Such form of guidance were called *judgmental statements* by Murmann et al. [111] and can be used to nudge users towards action. However, P3 would also like to ignore such warnings, if she finds the communication justified: "And then you can say – Okay, no, I don't trust that anymore. Or I always trust it, because I know that's good, even though the tool says otherwise" (P3). Murmann et al. already identified asserting users' own preferences as a gap in existing tools [111].

In her details view she focused on visualising *what*, *when*, and *where* data is transferred, without, however, being interested in the exact connection details: "I don't wanna know, or perhaps... but I don't think it's that important to know how much, and where – the connections" (P3). Interestingly, she was also the only participant, except P8, who included a visualisation of the type of data, arguing it could help her discover unwanted behaviour and patterns of her devices, like her printer unknowingly uploading all her documents to the Internet:

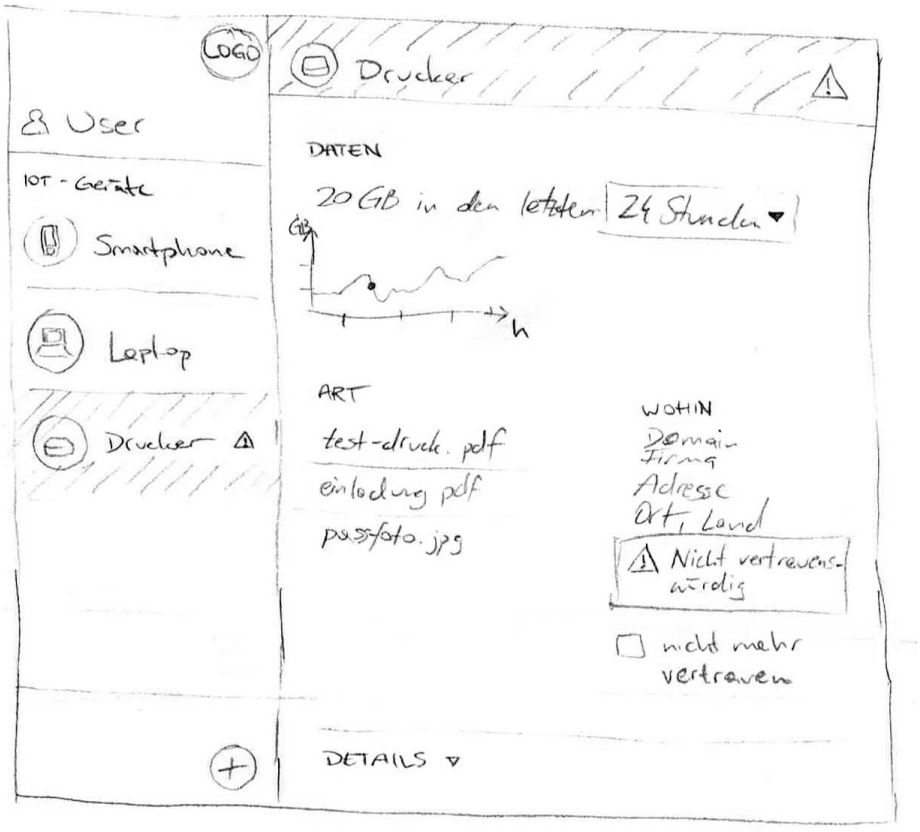


Figure 5.4: P3's sketch, including several warning signs and highlights in order to effectively communicate potentially unwanted information. In her sketch she also focused on visualising *what*, *when*, and *where* data is transferred, without being interested in exact connection details.

> Here it says [laughing] I'm not sure, maybe that's a bit silly, but *what* has been transferred and *where*... like domain, organisation, street, place, country [...] Like an address. Or who is behind that. [...] Maybe then you'll see a pattern, like the printer sending all printed JPEGs there...[P2: Or bank account details, picture of your credit card] Exactly. (P3)

P1 further highlighted potentially unwanted information in his mind map-inspired design, as seen in Figure 5.5, which includes a short explanation of the purpose of the contacted servers: "A useful information would be a short description of the purpose of this connection: 'This server is used to transfer operating system updates for macOS' or something like that. [...] Just so you know what's going on in general" (P1). To further improve the overview, he imagined classifying servers as *useful* and *useless*: "Grouped, first those concerning only functionality – something that is useful to the user. And then

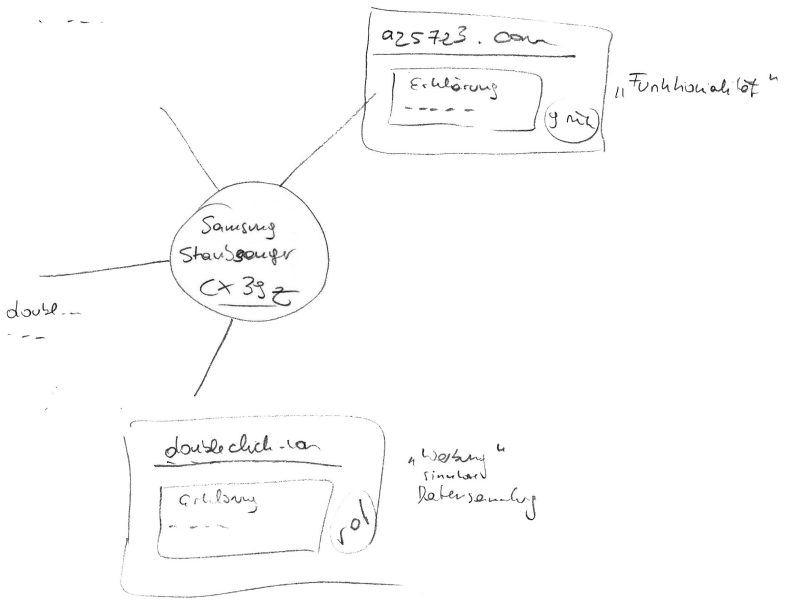


Figure 5.5: The first variant of P1's sketch showing colour coded servers in a mind map-inspired design, differentiating between *useful* and *useless* ones, and providing additional short descriptions about their purpose.

the others like 'advertising,' 'data collection' – the useless ones" (P1). Further colour coding these categories as green and red, respectively.

P1 also designed an alternative version of his sketch, as seen in Figure 5.6, which was inspired by Little Snitch's connection list (see Figure 3.1), where he similarly added the already mentioned classification and colour coding, with the explanations now only shown on demand on the right-hand side after clicking on a server name. Finally, he also borrowed the blocking controls from Little Snitch, as he found them easy to understand: "Kind of like in Little Snitch. That was a nice and simple solution with these crosses and check marks. That was actually really good. Very subtle and small" (P1).

P5 similarly created a design that focuses on a quick overview of potentially unwanted device behaviour, without having to manually dig into the details. To display such overview, he envisioned a graph-like dashboard in Figure 5.7, which includes a node for each device, with a colour coding and summary of its activity in the attached card:

> A dashboard that shows what is going on. Where it, for example, says "Alexa → Audio; green, check." It's quite plausible that Alexa sends audio to Amazon. [...] But when it says, Alexa sends audio recordings to Samsung, then that's very strange, and this box would be red for me. Because why should Alex send audio to Samsung? [...] Or also, why would a TV upload large amounts of data? I could better understand why Alexa would do that. (P5)

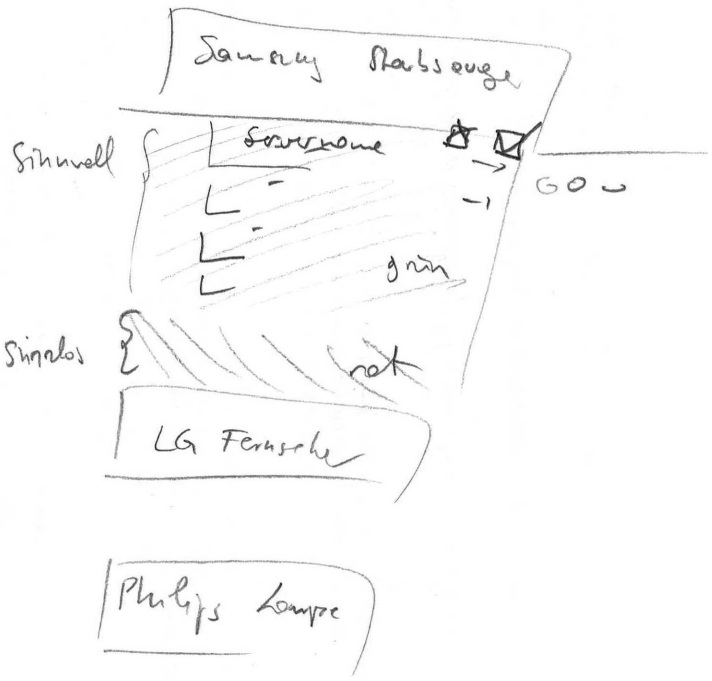


Figure 5.6: The second variant of P1's sketch inspired by Little Snitch's connection list, where he equally differentiates between useful and useless communication, now with additional blocking controls.

In addition to the overview, he also imagined a detailed visualisation of a device's communication behaviour after clicking on a device (see Figure 5.8), showing the contacted organisations as additional nodes:

> Then I would imagine that if I pick one node, all the others would disappear... Here I have a local node on one side and then connections to the nodes it was communicating with on the outside. [...] With information about the endpoint; who is that actually, and I'm not interested in the IP address on this level, but rather: is this Google, etc... (P5)

Furthermore, zooming in would provide additional details about the transferred data and their frequency:

> And then I can zoom in on the next level of detail, so to speak. [...] And on this level I can see what has been sent there. Already on a classification level. For example, would this rather be classified as microphone recordings, or as video, or just binary information like "switched channel," which is sent periodically. (P5)

Figure 5.7: The dashboard-like overview of P5's sketch, containing a node for each device, with a summary and colour coding of their behaviour in the boxes below them. This should allow users to quickly detect potentially unwanted behaviour without having to dig into the details.

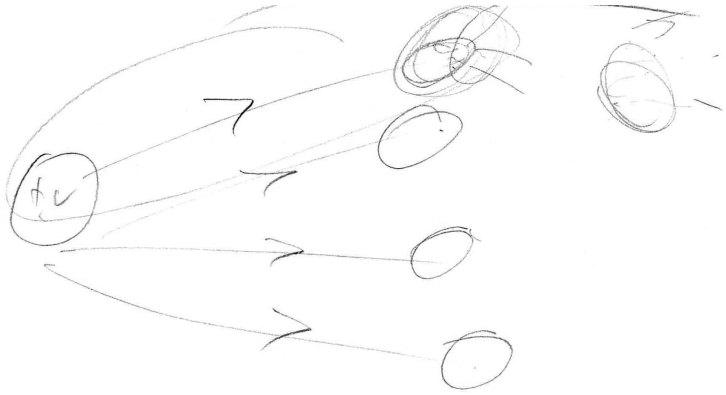


Figure 5.8: The details on demand in P5's sketch, visualising the contacted organisations of a single device after clicking on a node in the overview. Zooming further in would provide additional information about the type of data transferred to each organisation.

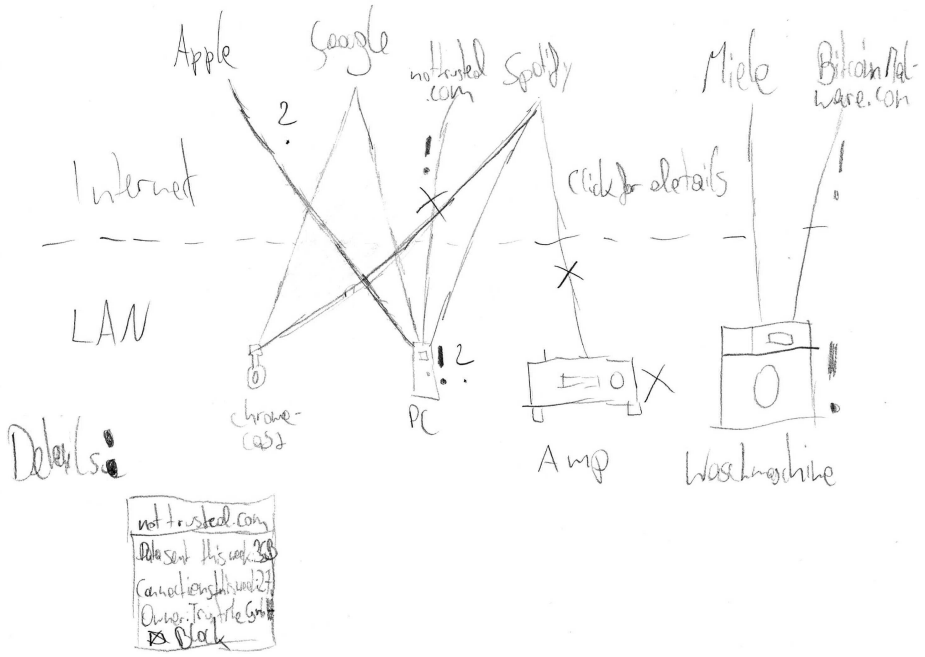


Figure 5.9: P4's sketch, showing local devices contacting various service providers. He used exclamation and question marks both on devices and on individual connections to mark potentially unwanted behaviour and communication. By clicking on a connection, further details can be shown on demand, offering the option to block it, as visualised through the crossed lines.

5.3 Group by Organisations

As already discussed in Section 4.1, most participants preferred to know the contacted organisation rather than low-level information like domains or IP addresses, a visual encoding Murmann et al. [111] called *service providers*. Figure 5.9 shows how P4 realised such visualisation, which contains his local devices, distinguished by device names and icons, connecting to various service providers on the Internet. To aid the discovery of potentially unwanted communication, he added both exclamation and question marks, not only to the specific connections but also to the affected devices, signalling to the user that further investigation is required.

However, while P4 focused on organisations, he realised during the discussion that he used domain names for malicious services like `bitcoinmalware.com`, which might, however, be a suitable alternative in situations where an owner is not easily identifiable. Clicking on a connection further opens a detail window, as shown in the lower left-hand side of Figure 5.9, containing statistics like "Data sent" or "Connections this week," and additionally allows users to block the communication per device, which is signalled by the cross icons on the connections.

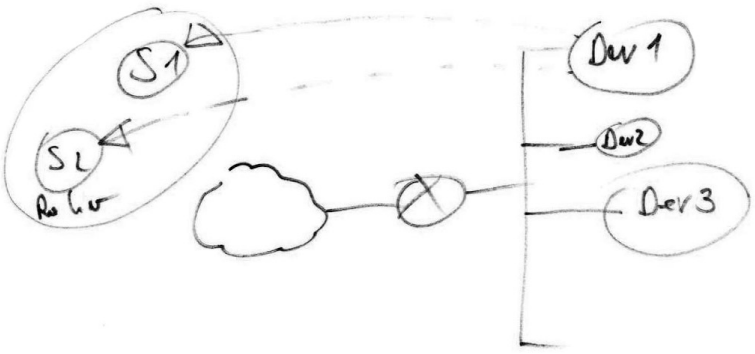


Figure 5.10: P12's sketch, showing how his device contacted two servers that are operated by the same provider, as indicated through the named lasso around them. The size of the devices also indicates their relative amount of transferred data.

P12 similarly visualised the contacted service providers in his sketch (see Figure 5.10), which was inspired by a router-like topology diagram, featuring local devices on the right-hand and the contacted external entities on the left-hand side. To illustrate the relative amount of data transferred, he used the size of the local devices and the width of the connection lines, with "Device 3," for example, having transferred the most. During the discussion he further added that the cloud in his sketch actually serves no real purpose and was just intended to distinguish between local and external entities.

An important design element in his sketch is the circle around the contacted servers, which groups them by similar purposes or the same service providers: "Host names and then a lasso around these two servers to indicate that they belong together. Indicating, for example, that they belong to the same provider or something" (P12). In his sketch he used the name "Radio" to group servers providing the main functionality for his Internet radio. This grouping further allows him to quickly see whether unrelated and potentially unwanted servers are being contacted by his devices, as they would be prominently placed outside the group. He also imagined colouring them in red, if they are known to be malicious: "And if it's additionally known that the server is somehow malicious (in the sense of a blacklist), then it flashes red or is coloured red with different intensity" (P12).

5.4 Geographic Visualisation

With the similar goal to quickly identify unexpected data transfers, P2 created a geographical visualisation of his devices' communication behaviour in Figure 5.11, which, for example, shows his smart iron (the rightmost device) communicating with a lot of Russian servers:

> My intention was rather to find out which device communicates where, at what strength. [...] For example, if I look at my culprit, my iron, it communicates

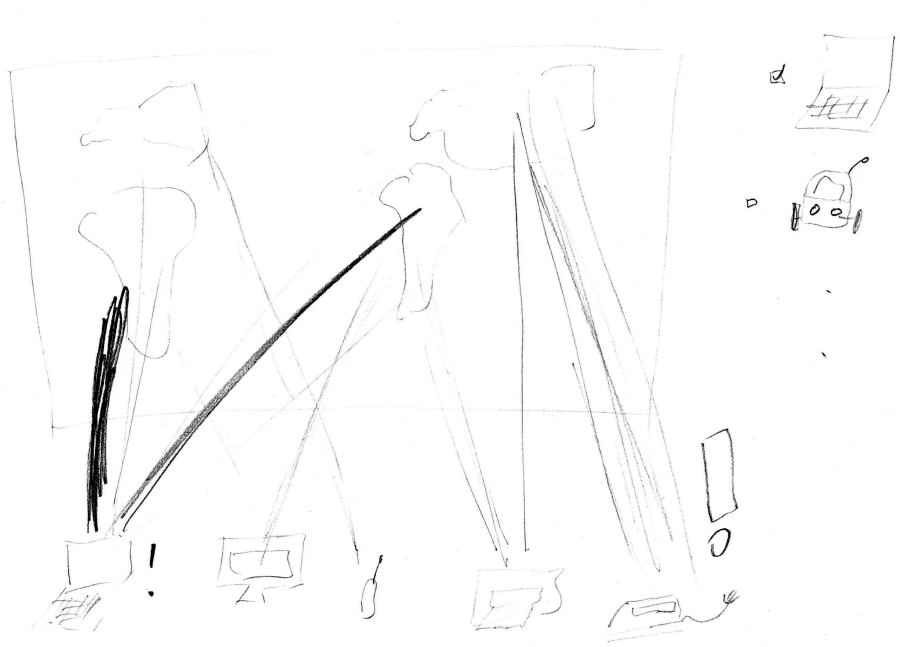


Figure 5.11: P2's sketch, providing a geographic visualisation of his devices' communication behaviour to quickly highlight unexpected connections, like his smart iron opening a lot of connections to Russia. He further used exclamation marks to alert users to potentially unwanted behaviour.

> quite a lot with Russia [laughing]. The idea here is to get a quick overview whether a device opens some connections that you don't expect. (P2)

To prevent certain devices from cluttering the view, he added the ability to hide selected ones through the controls on the right-hand side, so that potentially unwanted communication is not obscured: "[With all devices] this is flooded rather quickly. Because a laptop simply tends to communicate much, much more than an iron. And then unwanted things hide in the masses again" (P2). To further aid the discovery of potentially unwanted communication, he similarly used exclamation marks on his devices to signal that some action may be required by the user. Clicking on a connection line shows more details about the contacted servers, with the ability to block unwanted communication.

5.5 Time-Based Visualisation

In comparison to other participants, P7 put more focus on visualising the communication behaviour over time in his sketch (see Figure 5.12), with particular interest in knowing whether his devices are communicating right now, distrusting any communication without explicit user interaction (see Section 4.2.2):

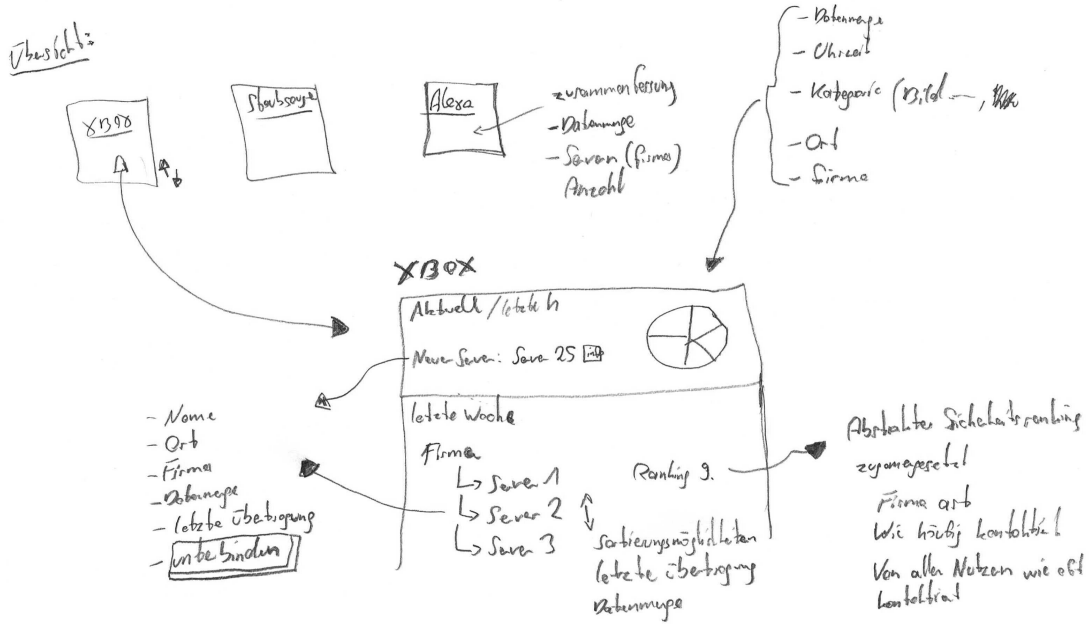


Figure 5.12: P7's sketch, distinguishes between what his devices communicate right now and what happened before, and further uses an abstract security ranking to highlight any potentially unwanted communication.

> Some kind of overview where I can see my device. One box would be a device, my vacuum for example. And that I can somehow see, ok, it's transferring data *right now*. Because that would be interesting, if I sit here [during the interview] and see that my vacuum transfers data [while it's not in use], that would be kind of weird. (P7)

Such device activity is indicated through the up down arrows right next to the device cards, which further contain a short summary of a device's communication behaviour, including the amount of data transferred and the number of contacted organisations, providing further hints of potentially unwanted behaviour.

Clicking on a device opens its details view, which is visible in the lower part of his sketch, visualising the contacted domains and organisations, again split by what is happening right now, or during the last few hours, and what happened before: "So for me, the last week would probably be the most important... I find it really important that there is always a split between what is current, and what has happened" (P7).

By focusing on time, the tool can further highlight new information so that users are not required to manually go through all details, which would further allow him to detect changes after automatic device updates: "But that would be quite interesting – functions

that get added. So if I buy a product and it doesn't have a Facebook connection, and then somehow it updates itself, which you may not even notice, and suddenly it has a Facebook connection" (P7).

Similar to other participants, P7 also included a mechanism that highlights potentially unwanted communication by using an abstract security ranking consisting of several parameters like the organisation's business model, the frequency of communication, and if others have already contacted the server before:

> For example, an abstract ranking that maybe consists of the type of organisation. Where you say, ok, that's an advertising company, maybe that's not so good. [...] And then how often it's contacted... how many *others*, in the sense of other users – whether I'm, for example, the only one, the first one that sends something there, or not. (P7)

Such ranking would allow him to focus only on potentially unwanted communication, without having to manually check everything:

> [Server with ranking "$C-$" or "9"] That's strange, or let's say, maybe something I should have a closer look at. On the other hand, there is no point if everything is, I don't know – I click on the device and everything is just ranking A, everything is great, green, all cool – then I don't have to click through all the servers. (P7)

However, he further wishes to manually influence the ranking algorithm by whitelisting organisations he trusts, such as Google, despite its advertising-related data processing, which reflects users' personal preferences, as already discussed in Section 4.1:

> Where I can say, okay, I have a few organisation I generally trust. Or maybe not generally... where I say, okay, I don't mind sending data to, for example, Google, that's actually OK. [...] So just because Google is an advertising company and I send lots of data to them, that not everything is suddenly ranked badly, because it's not... or because I don't really mind, if I send data to Google. (P7)

Finally, P7 also desired to manually decide which server to block based on the provided information, such as the country of origin, the description, the last time of communication, and the amount of data, as already mentioned in Section 4.5.5.

P6 similarly focused on a time-based visualisation and created a report-inspired visualisation (see Figure 5.13) that features a stacked area chart showing the distribution of the transferred data for each device, with a time series for each contacted organisation. However, compared to P7 he was not interested in checking the results in real-time, but preferred a monthly report to find and investigate any outliers: "Such a monthly graphic

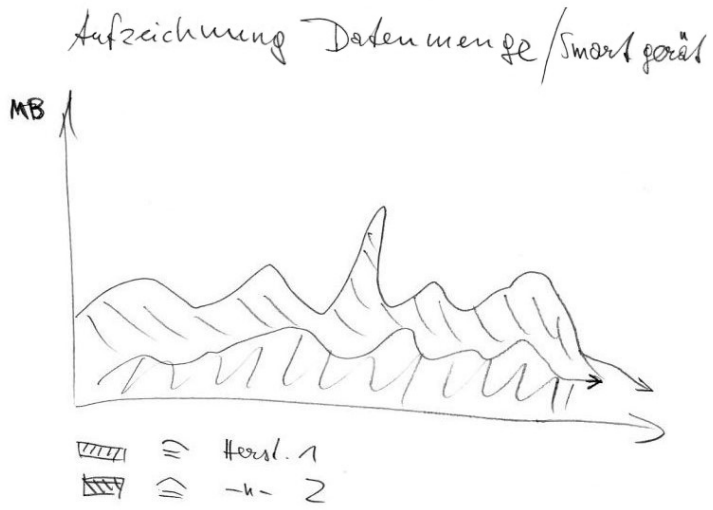


Figure 5.13: The first part of P6's sketch, showing the distribution of the contacted organisations over time in a stacked area chart. Compared to other participants, he was not interested in real-time visualisations, but preferred to receive a monthly report that he could investigate in detail.

would certainly be informative and manageable. Once a month it's reasonable to take time and review the results of such a monitoring" (P6).

Inspired by a radial histogram, P6 further created a second visualisation for his report (see Figure 5.14) showing the number of contacted organisations for a device, with the length of the bars proportional to the amount of data transferred. This would allow him to quickly see just by the shape of the diagram, if an unexpected amount of data is transferred, or an unexpected number of organisations are contacted – in the best case only one organisation, the manufacturer of the device: "Where you can see – oh, whoops, this star looks rather wild, and on this one I only have two or three, in the best case maybe only one [...] Then you see it's going reliably only to the manufacturer, and not some, I don't know, third or fourth parties" (P6).

To support the discovery of potentially unwanted organisations, he similarly added a colour coding and icons that indicate the availability of additional information about potential risks: "That you might highlight this with colour – attention, that's a connection you should investigate or take action – as warning or hint" (P6). Furthermore, he wished to block unwanted communication to affected organisations: "And then perhaps even block it, that would of course be... if that's possible. To put it on a blocklist or something. That you can say, I'm blocking that for the device, that's not appropriate, not the purpose of the device" (P6).

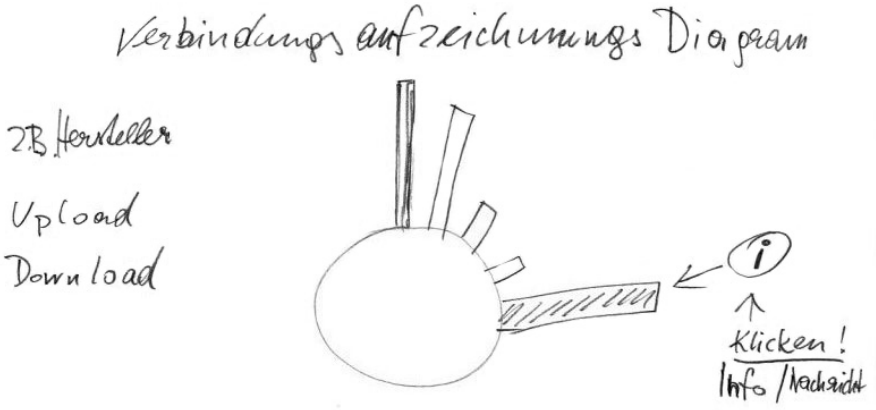


Figure 5.14: The second part of P6's sketch, showing the number of contacted organisations through the bars of a radial histogram-inspired visualisation, with the length of the bars proportional to the amount of transferred data. Furthermore, icons and colour codings provide additional hints and warnings about potentially unwanted behaviour.

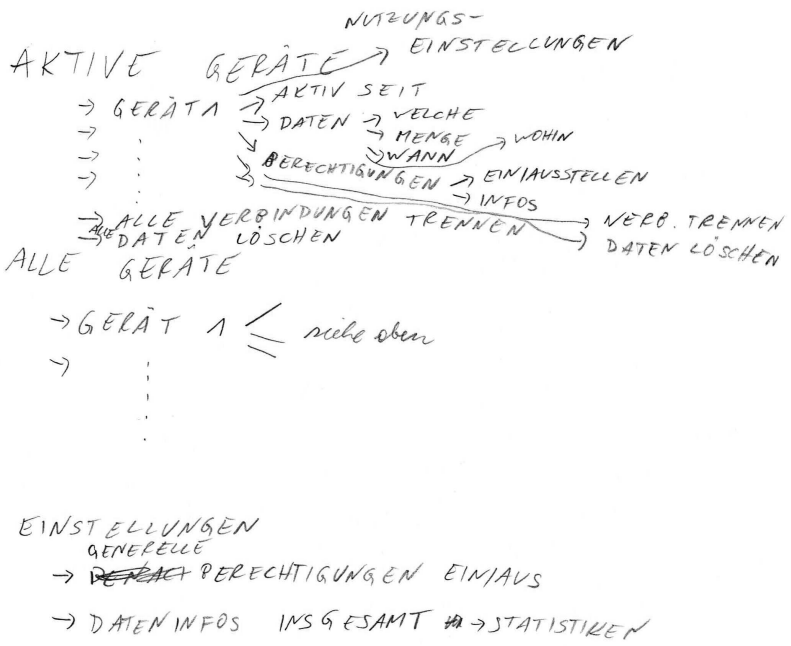


Figure 5.15: P11's sketch, where he primarily imagined a single interface to control the behaviour and permissions of all his devices, where he could, for example, prevent all his devices from using the microphone or from sending GPS data.

5.6 Focus on Control

P11 chose a different approach and instead of visualising the communication behaviour, he focused on a permission-based system (see Figure 5.15) that allows him to control the behaviour and permissions for each of his devices through a single interface. His tool would directly alter a device's configuration and prevent any unwanted communication or resource usage, or even completely block a device's communication, even if this would cause it to stop working, as already mentioned in Section 4.5.3:

> There you can set all of a device's settings... for example, when the vacuum should be running or such things. [...] And there you can see and set the permissions for each device. Even globally for all, where you can say, no device should send GPS data, or no device should record audio. (P11)

However, should a device require a certain permission, he imagined a notification that would warn him in such situations, without, however, preventing such decision: "And maybe some kind of notification – this device can't work without this permission. Then you have to deal with that separately" (P11).

In addition to controlling the behaviour, he also envisioned to delete any collected data from his devices as well as from any remote locations, even though he was aware that this would probably not be possible: "And maybe delete all the data it stored locally... the other kind probably won't work" (P11).

P8 similarly focused on controlling device behaviour through granular options, with her design being inspired by a smartphone app update dialog (see Figure 5.16) that provides a prompt detailing what type of data would be collected, for what purpose, and which organisations would receive it – and give users fine-grained control over what they want to accept, like allowing voice recordings to be transferred to certain organisations, but not to others. Interestingly, she also explicitly added what information would not be used or shared.

> I simply imagined what the app could say [laughing] [...] What kind of information it shares, why, and where; and then whether you actually want that [...] That you can simply tick what exactly you want to allow from all that, and what not. [...] For example, this app wants to... or with this update the app is able to share at what time you use it. And that you can then decide if you want that, or not. [...] But yeah, this is a very simplified form of how I imagined such tool could work. (P8)

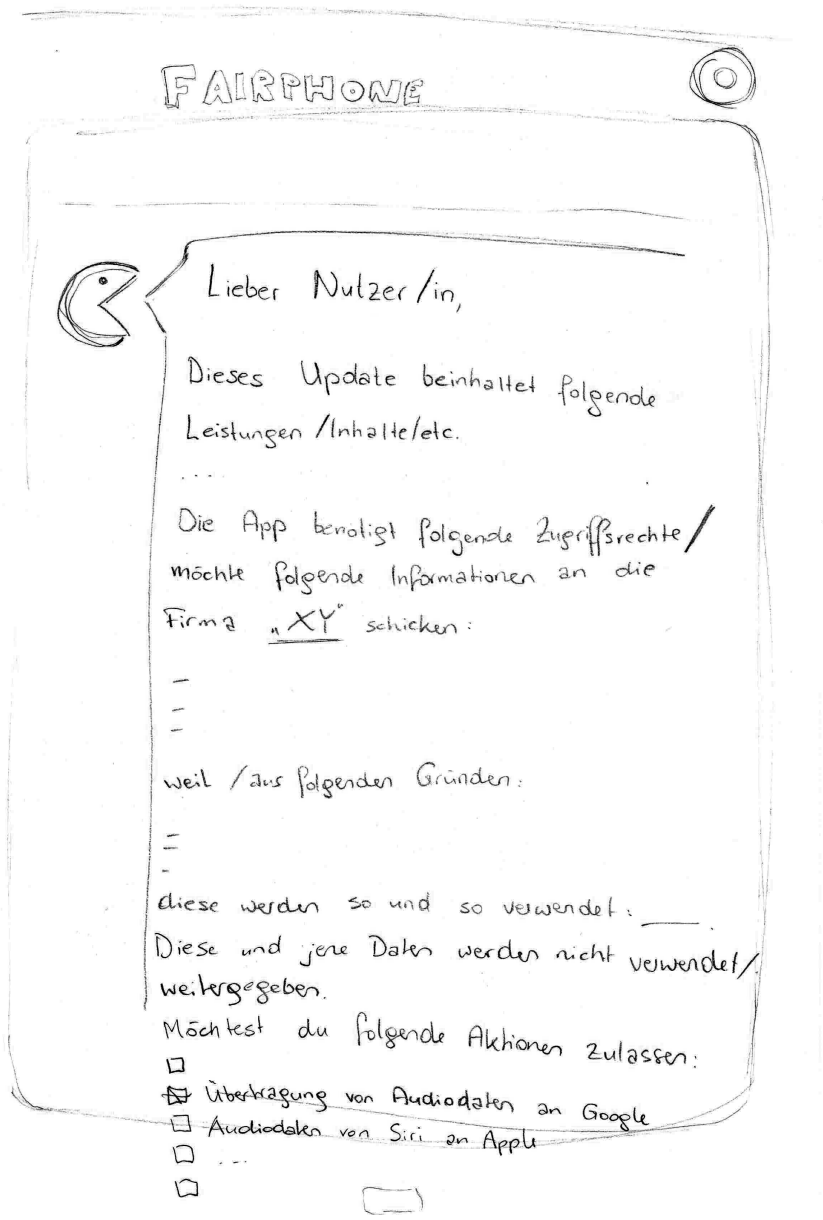


Figure 5.16: P8's sketch, showing a text-based visualisation containing details on what kind of information the app or device shares, why, and where; with the ability to decide per check boxes whether you actually want that.

Part II

Implementation and Evaluation

CHAPTER 6

Providing Transparency and Control over Devices' Communication Behaviour

The previous chapter showed how participants imagined a usable visualisation of their devices' communication behaviour. This chapter discusses the implementation of a tool, hereinafter called IoT Guard, that can extract a selection of the desired information from the devices' network traffic in order to provide a visual encoding that is inspired by participants' sketches, as discussed in the next chapter. Both Chapters 4 and 5 identified contacted organisations as a primary abstraction. The following sections therefore give a short introduction into the concept of network layers; what challenges a passive analyser faces, including the challenge to capture the desired network traffic; which tasks IoT Guard has to perform in order to detect the contacted domains and ultimately their associated organisations; and finally, how to use them to block unwanted communication in a usable way.

6.1 Network Communication Basics

To extract information exchanged by applications over the network, IoT Guard first needs to understand how such information is encapsulated and processed by the different *network layers* involved. These layers divide communication tasks into different concerns, with each providing a fixed set of features, while using the interfaces of the ones below. Figure 6.1 gives an overview of the layers involved in the TCP/IP model and shows how messages are "vertically" passed through each layer, with each adding or removing its own control information as header and optional footer – a process called *multiplexing* for the sender and *demultiplexing* for the receiver. The message of an upper layer is referred to as the *payload* of the underlying layer, while header and payload combined

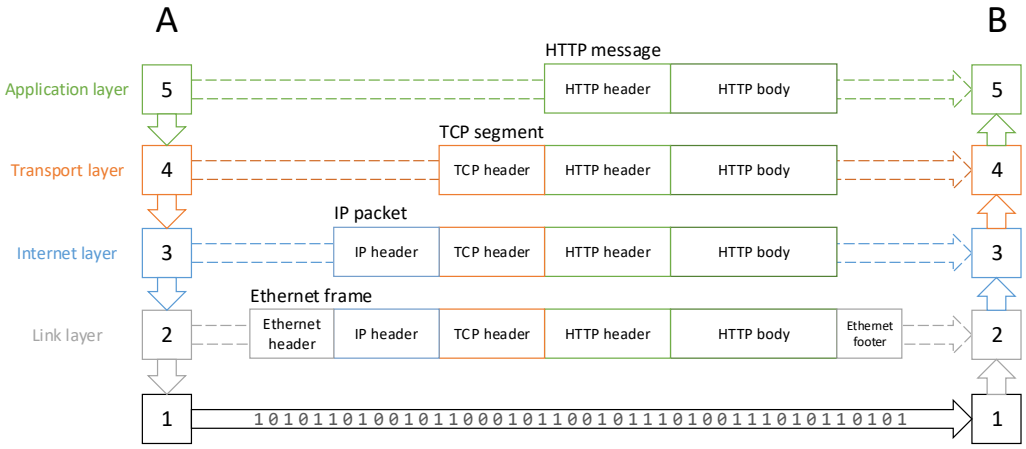


Figure 6.1: Network layers and message encapsulation in the TCP/IP model (after [85]). In this example a HTTP message is sent from device A to B. Note that the actual data transfer only happens on layer 1, the network hardware. The dashed horizontal arrows only symbolise the logical connections between the same layers.

are generally called *packet*, but as Figure 6.1 shows, each layer has their own naming conventions for their exchanged messages [85].

In addition to encapsulating messages, network layers may also fragment them into multiple packets, if the payload is larger than the maximum message size of the network. This fragmentation is the basis for *packet switching*, the Internet's main communication method, where each packet is sent independently and without a predefined route through the network. However, this might result in packets arriving out of order, being lost, corrupted, or further fragmented into smaller packets. It is the responsibility of the receiver to correctly reassemble the packets again, and to request retransmission of missing or corrupted ones [85].

6.2 Passive Analysis Challenges

While the passive analysis of network traffic, i.e., without actively taking part in the communication, is comparable to the demultiplexing task of the receiver, several additional challenges arise, the first one being able to actually capture devices' network traffic, as in switched networks the traffic is only routed to the intended recipient. However, there are several ways how a passive analyser can still get access to the desired network traffic, by using, for example, dedicated network hardware like network taps, or management features like a mirror port of compatible switches [139]. To minimise the configuration overhead, IoT Guard uses another method called *Address Resolution Protocol (ARP) spoofing*, or ARP cache poisoning, which manipulates routing information at the Internet layer to impersonate the network's default gateway, and trick devices into communicating with IoT Guard instead. IoT Guard then analyses the devices' network traffic and

forwards it to the actual gateway, putting the tool in a Man-In-The-Middle (MITM) position [25].

To realise such ARP spoofing for selected devices, IoT Guard reuses the *Apate* component of the Raspberry Pi-based Upribox. While the Upribox would provide a number of privacy enhancing features like automatically blocking trackers and removing Personally Identifiable Information (PII) from the network traffic [33], IoT Guard disables anything that modified the network traffic in order to obtain an unfiltered view of the devices' actual communication behaviour, while still providing users the option to *manually* block unwanted communication based on personal preferences and gained insights, as participants desired in Section 4.5.5.

A passive analyser has to additionally deal with so-called *packet drops*, which can occur if the tool can't keep up with the number of packets arriving at its network interface. While packet drops are an intended mechanism of the Internet Protocol (IP) in dealing with network congestion, a passive analyser can't request retransmissions, like the Transmission Control Protocol (TCP) normally would [124], and therefore must be able to handle gaps in the reassembled data stream, which may cause control information like the initial handshake or teardown of a session to be lost [22]. Analysing the network traffic of multiple devices further increases the importance of an efficient resource management, including various timers to release expired resources. While a separation of data capture and analysis could reduce packet drops due to increased capture performance, this would introduce the additional challenge of managing the storage of the captured packets [10], and would further prevent a real time visualisation, as participants desired in Section 5.5. For these reasons, IoT Guard analyses the captured traffic on the fly.

6.3 Analyse Outbound Communication

With access to the network traffic, IoT Guard uses the abstraction of *Network Connections* (see Figure 6.2) to analyse any outbound communication, and perform the aforementioned demultiplexing task by reassigning packets to temporary data exchanges between two application processes, which are identified by their IP addresses and ports, and are also called *socket pair* in combination [85]. A Network Connection is created for the first IP datagram seen for a socket pair, where the `protocol` field determines the encapsulated transport protocol, and the corresponding ports the respective application protocol. TCP 80, for example, corresponds to the Hypertext Transfer Protocol (HTTP) [71]. To delegate further analysis tasks for each layer, IoT Guard assigns each connection a chain of analyser classes matching its network protocols, which are described in detail in the remainder of this chapter. To determine the end of a connection, IoT Guard either uses the explicit close provided by the *TcpAnalyser* (see Section 6.4.2), or a timeout due to inactivity. This core architecture is closely inspired by the open-source Network Security Monitor (NSM) "Zeek" (formerly "Bro"), which provided a reference for a flexible implementation of the analysis tasks, including an efficient timer management and gap detection mechanism [120, 148].

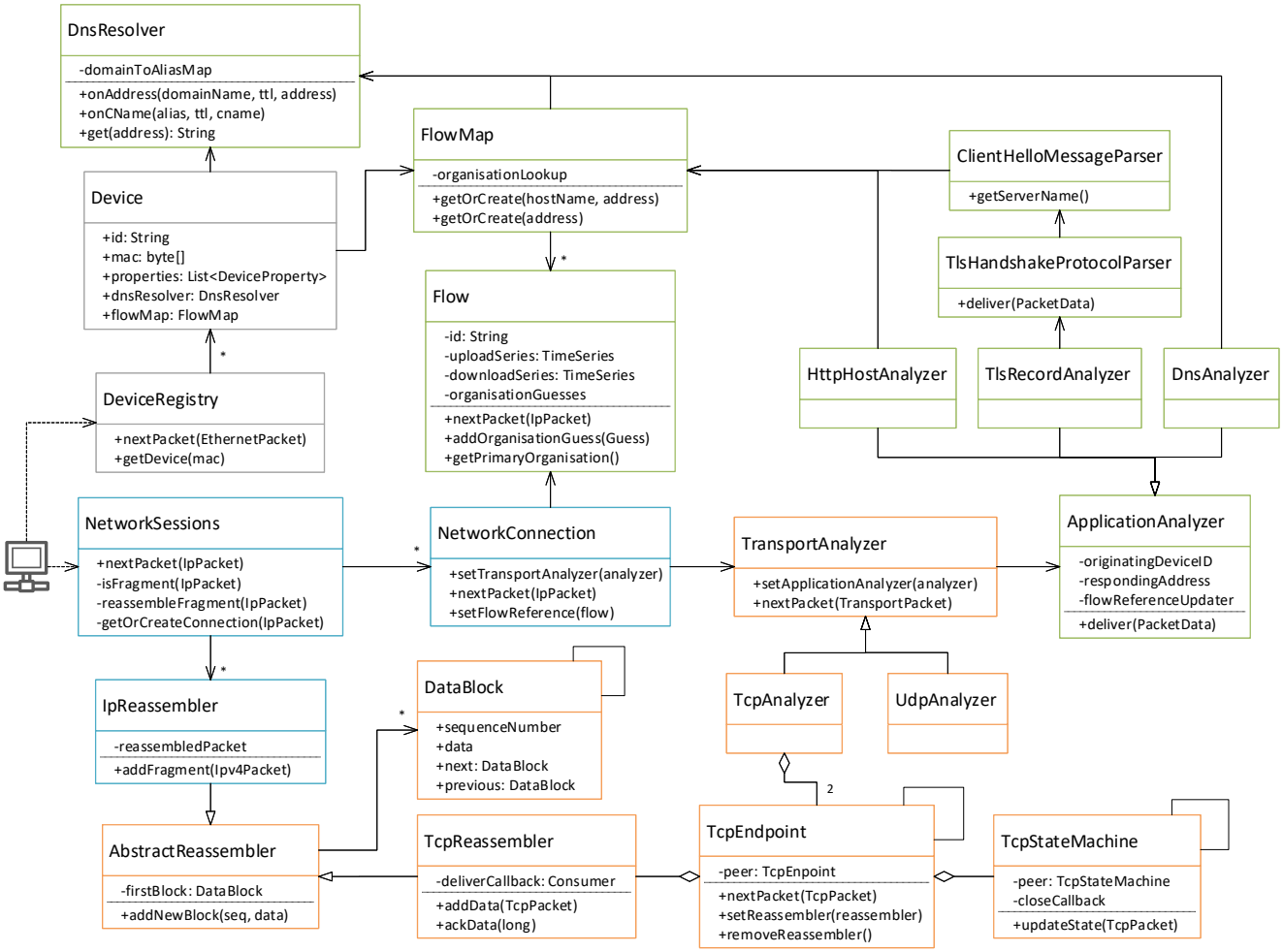


Figure 6.2: Simplified class diagram for reassembling the data stream, extracting contacted domain names, and managing Flows. The flexible architecture was inspired by "Zeek" [120], and allowed the author to quickly iterate and deploy IoT Guard in real-world networks. The colour coding is equal to the network layers in Figure 6.1.

As Network Connections only represent a temporary data exchange between two application processes, IoT Guard uses the additional abstraction of *Flows* to keep a record of contacted hosts for each device. Flows are identified by their (domain) names and collected in the device's *FlowMap* (see Figure 6.2). Furthermore, a Flow reference is assigned to its corresponding Network Connection in order to track the temporal distribution of transmitted data and communication-related details like ports or requested resources (see Section 6.5.1). However, before any Flows can be created or assigned, the contacted domain names must first be extracted for each connection, which in turn requires a reassembled data stream.

6.4 Reassemble Fragmented Data Stream

To analyse data transferred on the application layer, the original data stream must first be reassembled for both IP fragments and TCP segments.

6.4.1 IP Reassembly

While both IPv4 and IPv6 support IP fragmentation, IoT Guard currently only supports the reassembly of IPv4 fragments, which uses the following header fields [123]:

- `Identification`: A unique identifier per IP datagram, which is shared among all fragments.
- `Fragment offset`: The position of the fragment inside the original datagram, which is used to reassemble the fragments in the correct order again.
- `MF` (More Fragments): A flag set on all fragments except the last, which determines the total length of the original datagram.

Using the condition `fragmentOffset != 0 || MF`, IoT Guard can identify if a packet is part of a fragmented IP datagram. Reassembling its fragments is then the simple task of ordering their payloads and resolving any potentially overlapping data, until all fragments have been received [85]. To account for potential packet drops (see Section 6.2), IoT Guard releases any resources, if not all fragments were received before a timeout. After the original datagram has been reassembled, it is forwarded to its corresponding Network Connection.

6.4.2 TCP Reassembly

Reassembling the TCP data stream is a bit more complex, because it requires a continuous delivery of reassembled data to the application analyser, as, in contrast to the fixed total length of reassembled IP datagrams, TCP has no knowledge of message boundaries, and instead just transfers application data in a continuous stream of data, fragmenting it into TCP segments [124]. To avoid additional IP fragmentation, a method called Path Maximum Transmission Unit (MTU) Discovery is used to determine the maximum TCP segment size of the internetwork [105].

Similar to the *fragment offset* field of IP fragments, TCP uses *sequence numbers* to uniquely identify the relative position of each transmitted byte in the data stream. To ensure a reliable data transfer, the cumulative *acknowledgement number* acknowledges the gapless receipt of data up to the specified sequence number [124]. The *TcpReassembler* uses these acknowledgements as a signal to forward any reassembled data up to the acknowledged position to the application layer. If the data still contains gaps due to packet drops, IoT Guard skips over them and reports their occurrence to the application analyser, which might decide to abort further reassembly all together, or perform some

other error handling. Gap awareness allows IoT Guard to reduce resource consumption by eliminating the need to wait for dropped segments and being able to immediately remove all cached segments.

As a further performance improvement, IoT Guard only selectively reassembles TCP streams that are of interest to the available application analyser. For HTTP, for example, IoT Guard only processes the stream originating from the client in order to extract information from its HTTP requests, while safely ignoring any HTTP responses. Likewise, IoT Guard completely stops the reassembly of Transport Layer Security (TLS) records after the initial handshake has been processed, as no further information can be extracted from the following encrypted application data. Unsupported protocols are ignored altogether.

While not strictly necessary for reassembling TCP segments, IoT Guard also manages the state of TCP sessions (see Figure 6.2) to immediately free resources on connection teardowns [124], and additionally uses an inactivity timeout to account for packet drops that might cause IoT Guard to miss the active close.

6.5 Extract Contacted Domains

To identify connection-independent Flows, IoT Guard extracts the contacted domain for each Network Connection using two different approaches based on the encountered application protocol: for HTTP and TLS, IoT Guard uses protocol specific fields to extract the domain directly; for any other protocol, the previously captured Domain Name System (DNS) messages are used for an indirect lookup. The following sections describe the different approaches in detail.

6.5.1 HTTP Host Header

For unencrypted communication over the text-based HTTP, a straightforward way to extract the contacted domain is to parse HTTP request messages and use their *Host* header values. The first line is also called the *request line*, which contains the HTTP method, target, and version. To extract request messages from the continuous data stream, IoT Guard parses the stream line by line, waiting for a request line, and if found, adds each subsequent line as a header to the request, until the first empty line is read, marking the end of the header fields [44]. While IoT Guard does not parse any potential message-body, it additionally stores the requested Uniform Resource Identifier (URI) in the corresponding Flow, and tracks the amount of uploaded data and its time of occurrence, as shown in Section 7.3.1. To improve the readability of the URIs in the user interface, they are further URL decoded to reverse any percent-encoding, as seen in Listing 1, where two decoding passes were needed to fully remove any nested encoding [11].

```
0: adproxy?u=http%3A%2F%2Fad.com%2Fad%3Ft%3D%257Be%253D1%257D
1: adproxy?u=http://ad.com/ad?t=%7Be%3D1%7D
2: adproxy?u=http://ad.com/ad?t={e=1}
```

Listing 1: Reversing the nested percent-encoding of a URI target, in two passes, greatly improving its readability.

6.5.2 TLS Server Name Indication (SNI)

For encrypted connections using TLS, IoT Guard attempts to extract the contacted domain from *Client Hello* handshake messages containing a SNI extension, which is used to support TLS on virtual hosts, i.e., servers that host multiple web services accessible through different domain names. Similar to TCP segments, TLS adds a layer of encapsulation through its TLS records, which are used by various TLS protocols, such as the handshake protocol, which exchanges unencrypted control information at the beginning of a secure connection, containing the aforementioned initial Client Hello message [14, 29].

6.5.3 Domain Name System (DNS)

As a final source IoT Guard also extracts `A`, `AAAA`, and `CNAME` Resource Records (RRs) from captured DNS messages, and adds them to the *DnsResolver* class of the corresponding device, which not only provides a lookup from an IP address to its domain name but also searches for the last alias defined by `CNAME` records. To give an example, consider the captured DNS RRs in Listing 2. If the device now opens a connection to `104.83.4.160`, not using HTTP or TLS, the DnsResolver first checks its address records, returning `a1363.dscg.akamai.net`, which is further recursively resolved to the last alias found in its `CNAME` records, resulting in `crl.microsoft.com`, the domain the device originally made the DNS request for. Considering these alias entries allows IoT Guard to provide more specific results, which are especially useful for web services hosted on Content Delivery Networks (CDNs), such as Akamai in this example. While IoT Guard also supports active reverse DNS lookups as fallback, this would only result in `[...].deploy.static.akamaitechnologies.com`, which is less accurate, but still more insightful than the IP address alone, in case no matching RRs have been captured [103, 104].

```
crl.microsoft.com.       IN  CNAME   crl.www.ms.akadns.net.
crl.www.ms.akadns.net.   IN  CNAME   a1363.dscg.akamai.net.
a1363.dscg.akamai.net.   IN  A       104.83.4.160
```

Listing 2: DNS RRs for the domain “crl.microsoft.com” containing multiple CNAME records used to resolve the originally requested domain from the contacted IP address.

6.6 Identify Organisations and Subsidiaries

After extracting the contacted domain names and assigning connection-independent Flows to each connection, IoT Guard tries to lookup the associated organisation for each Flow. To simplify this lookup, Flows are grouped by their respective *root domains* using the Public Suffix List (PSL), also called "effective Top-Level-Domain (TLD) list," which defines the TLDs under which a user can directly register a name [109]. A root domain is then the TLD plus one, since all further subdomains very likely belong to the same owner, requiring only a single organisation lookup for each root Flow. To give an example, `my-app.s3.amazonaws.com` is considered a root domain, as `s3.amazonaws.com` is included in the PSL.

To lookup organisations and additional properties about them, IoT Guard uses several sources and assigns a granularity and reliability to each in order to resolve potentially conflicting results, and additionally create a heuristic organisation hierarchy. The following sections describe the general algorithm for looking up organisations; how conflicting results are resolved, normalised, and used for the organisation hierarchy; which sources IoT Guard uses; and how additional organisation properties are collected.

6.6.1 Lookup Algorithm and Heuristic Hierarchy

Since there is no single authoritative source for looking up the associated organisation for a domain, IoT Guard uses a heuristic approach by combining multiple sources through the concept of *OrganisationGuesses* that contain not only the organisation but also the granularity and reliability of its source, as defined by the author during the evaluation. For every new root Flow, the associated root domain is passed to the *OrganisationLookup* class, which asynchronously looks up organisations from the available sources, and adds their results to the Flow as guesses. The guess with the finest granularity and highest reliability determines the associated organisation of the Flow, which is dynamically updated on better guesses. IoT Guard uses the following, increasingly coarse granularities: `Service` < `Organisation` < `IP-Owner`. This, for example, prevents the coarse `Organisation` guess "Facebook" to override the fine `Service` guess "Instagram" for the domain `instagram.com`. Furthermore, to account for IP-based sources, an additional lookup is performed for each new IP address seen for a Flow.

As a side effect of multiple sources with different granularities, IoT Guard is able to create a heuristic organisation hierarchy that at best represents a subsidiary relationship, or otherwise groups organisations by their cloud services used. To illustrate how the algorithm combines different guesses into such a hierarchy, consider the following three guesses for the contacted domain `youtube.com`:

1. The first source returns an `IP-Owner` guess "Alphabet," and being the first guess, the organisation is simply assigned to the Flow.

2. The second source returns a finer `Service` guess "YouTube," which takes precedence over the first, and updates the assigned organisation of the Flow. At this point, the Flow adds any remaining coarser guesses to the *HierarchyBuilder* of its assigned organisation. In this example, the coarser guess "Alphabet" is added to the HierarchyBuilder of "YouTube," creating the hierarchy `YouTube` < `Alphabet`.

3. The third source returns an `Organisation` guess "Google," which is finer than the first guess, but coarser than the second. By repeating the addition to the HierarchyBuilder as described in step 2, the new organisation is inserted between the two previous ones, creating the final hierarchy of `YouTube` < `Google` < `Alphabet`.

It should be noted that this hierarchy creation is independent of the order of guesses, and unique across all devices and Flows, which allows IoT Guard to first create the hierarchy `YouTube` < `Google` with guesses for one Flow, and later create the same final hierarchy as above, if guesses for another Flow return `Google` < `Alphabet`. To improve the predefined reliability metric for each source, IoT Guard additionally counts for how many Flows the same guess appeared in order to group the largest number of related Flows.

Finally, IoT Guard also contains a list of predefined, well-known subsidiaries, which were manually collected and verified during the integration of existing tracker lists, as detailed in Section 6.6.4. Such predefined parents take precedence over any heuristics.

6.6.2 Name Normalisation and Alias List

Using multiple sources for looking up organisations led to the challenge that each can return different variants of the same name, which not only creates undesired organisation hierarchies but also prevents IoT Guard from merging organisation properties. One normalisation applied by IoT Guard is to remove well-known legal entities from organisation names like "LLC," "Inc.," or "GmbH & Co. KG." Each again with several spelling variants, including extra spaces, dots, parenthesis, dashes, etc. To handle this great variation, IoT Guard uses a best-effort approach by combining a list of known legal entities with predefined regular expressions, and further removing everything after a comma or dash. While these normalisations cover most legal forms and slight spelling variations, they are not able to handle organisation aliases or name changes like "Pocket" and its previous name "Read It Later." To resolve such cases, which were discovered during the integration of tracker lists, IoT Guard additionally uses a manually curated alias list, as detailed in Section 6.6.4.

6.6.3 IP Address-Based Lookup

As the IP address of the contacted host can offer some valuable insight into the contacted organisation, IoT Guard includes multiple sources for coarse `IP-Owner` organisation guesses, as briefly discussed below.

Autonomous System (AS) owner

A primary source for such guesses is the AS of an IP address, which groups addresses that are under the same administrative control [60]. While this might only return the Internet Service Provider (ISP) owning the address, bigger organisations like Google and Facebook have their own Autonomous System Numbers (ASNs), which provides a straightforward way to identify services associated with these organisations [69]. To lookup ASNs, IoT Guard uses the regularly updated MaxMind offline database [99].

Cloud Service IP Addresses

While AS organisation lookups provide a reliable way to identify the administrative owner of the contacted IP address, it is not possible to differentiate between services hosted on cloud platforms and services provided by the hosting organisation itself. For example, services hosted on Amazon Web Services (AWS) and Amazon-owned services. To overcome this limitation, IoT Guard includes multiple lists of IP ranges used for hosting purposes by major cloud platforms, which are briefly mentioned below:

- **Amazon** provides their AWS IP ranges in a straightforward JavaScript Object Notation (JSON) format, which can be downloaded from a fixed address [4].
- **Microsoft** similarly provides a list of IP ranges for their Azure cloud platform as an XML file, even though there does not exist a fixed address to update the list, requiring some additional manual action [102].
- **Google** uses a different approach and includes their cloud IP ranges in Sender Policy Framework (SPF) records, which can be looked up by using recursive `TXT` DNS queries [57, 167]. Listing 3 shows the initial query for the predefined domain, and how IoT Guard recursively resolves any `include` directives until all IP ranges have been extracted.

```
$ dig txt _cloud-netblocks.googleusercontent.com
"v=spf1 include:_cloud-netblocks1.googleusercontent.com \
include:_cloud-netblocks2.googleusercontent.com \
include:_cloud-netblocks3.googleusercontent.com ?all"

$ dig txt _cloud-netblocks1.googleusercontent.com
"v=spf1 include:_cloud-netblocks4.googleusercontent.com \
ip4:8.34.208.0/20 ip4:8.35.192.0/21 ip4:8.35.200.0/23 ?all"
```

Listing 3: Example DNS queries to lookup Google Cloud IP ranges from SPF records in order to differentiate between Google-owned services and services just hosted by Google.

6.6.4 Domain-Based Lookup

With the root domains of Flows, IoT Guard can provide more fine-grained organisation guesses using the sources detailed in the following sections.

WHOIS

WHOIS lookups are a well-known and straightforward way to look up domain ownership details [24], but even though the protocol itself is simple, IoT Guard faces several challenges, like finding the WHOIS server responsible for the domain in question, as each TLD has its own server with no standardised naming scheme. However, there is an official "Root Zone Database" maintained by the Internet Assigned Numbers Authority (IANA) that contains administrative information for each TLD, including its WHOIS server [70]. To avoid manually parsing this semi-structured database, IoT Guard reuses the existing `tld.json` file maintained by an open-source WHOIS client [18]. But even then, so-called *thin registries*, as used by the TLDs `.com` and `.net`, still just respond with a referral, requiring a second WHOIS request to the referenced server [73], as shown in Listing 4.

```
$ whois fb.com
Domain Name: FB.COM
Registrar WHOIS Server: whois.registrarsafe.com
Registrar: RegistrarSafe, LLC
>>> Last update of whois database: 2019-10-31T13:32:26Z <<<

$ whois -h whois.registrarsafe.com fb.com
Domain Name: FB.COM
Registrar WHOIS Server: whois.registrarsafe.com
Registrar: RegistrarSafe, LLC
Registrant Organization: Facebook, Inc.
Registrant State/Province: CA
Registrant Country: US
>>> Last update of WHOIS database: 2019-10-31T13:32:36Z <<<
```

Listing 4: Two WHOIS lookups with the first returning only a referral for "fb.com," requiring a second request to extract the associated organisation and its country of origin. Responses have been shortened to only include relevant fields.

Once the full WHOIS response has been received, the next challenge is to extract the registrant's organisation and its country of origin from the non-standardised, semi-structured format, which can be different for each WHOIS server [24]. For this, IoT Guard uses a best-effort approach by combining several regular expressions that cover the most common WHOIS formats encountered during development and in related WHOIS projects [18]. However, the extracted properties still need to be checked for possible

redactions, as the Internet Corporation for Assigned Names and Numbers (ICANN) issued a temporary specification that requires the redaction of certain PII, since the GDPR entered into force on May 25 2018 [72]. To ignore such results, IoT Guard includes a manually curated list of observed redaction patterns, including examples like "Redacted For Privacy," "Not Disclosed," or "Masking Enabled."

Finally, if lookups start to fail, because IoT Guard has been temporarily blacklisted by WHOIS servers due to too many requests, an exponential back-off strategy is applied, and to further mitigate such cases, IoT Guard caches any WHOIS results on disk, preventing repeated lookups on application restarts. However, it should be noted that WHOIS servers often prohibit automated, high-volume processing of their data, which may require additional optimisations to be fully compliant, which are, however, not further discussed in this thesis, as IoT Guard is not yet available to the public.

X.509 Certificates

In addition to WHOIS, IoT Guard also uses end-entity X.509 certificates as an additional domain-related organisation source, which are extracted from their corresponding TLS handshake messages [21]. The *Subject* and its *Subject Alternative Name* extension are the only relevant fields, as shown in Listing 5, which contain a X.500 Distinguished Name (DN) with the following attributes [21, 83]:

- `CN`: The *common name* of the subject containing the domain the certificate was issued for, although this has been deprecated, and the domains contained in the Subject Alternative Name extension must be used instead to define a certificate's validity [132].
- `O`: The *organisation* this certificate was issued for.
- `C`: The *country* the subject, i.e., the organisation is residing in.

```
Subject: CN=www.tuwien.ac.at, O=Technische Universität Wien,
         C=AT
Subject Alternative Name: DNS:www.tuwien.ac.at, DNS:tuwien.at,
                          DNS:technischeuniversitaet.wien
```

Listing 5: Example subject and alternative name extension of a X.509 certificate, used by IoT Guard to extract the associated organisation of the domains.

However, not all X.509 certificates contain an organisation, as *Domain Validated (DV) certificates* only validate that the subject has control over the corresponding domains, without any attribution to a legal entity [56]. To still benefit from the grouping of domains in DV certificates, IoT Guard performs WHOIS queries for each root domain, until the first organisation is found, which is then associated to all domains. As this

might introduce some false positives for certificates that are issued for third parties not directly related to the contained domains, like the CDN "Fastly" with its "Shared TLS Certificates Service" [43], IoT Guard heuristically checks whether more than 16 different root domains are contained in a DV certificate. This threshold was chosen based on real-world tests, but may not cover all problematic certificates found on the Internet.

Crunchbase

For fine-grained `Service` guesses, IoT Guard uses the Crunchbase Open Data Map (ODM), which can be accessed via an API, or through a downloadable CSV file for offline access. Listing 6 shows a shortened API result for Google, which contains, among other properties, a profile image, short description, and country of origin. The main entry point of the API is the `domain_name` query, which returns matching organisations based on their primary `domain` property. However, the query falls short for other related domains, like `google.at` instead of `google.com`. To overcome this limitation, IoT Guard additionally uses the `name` API endpoint to look up properties for organisations that have been identified by other sources, a mechanism further discussed in Section 6.6.5. Even though Crunchbase also contains subsidiary and alias information on its website, they are unfortunately not included in the free ODM, but where nevertheless used during manual researches while merging tracker lists.

```
{
  "type": "OrganizationSummary",
  "uuid": "6acfa7da1dbd936ed985cf07a1b27711",
  "properties": {
    "name": "Google",
    "short_description": "Google is a multinational corporation
    that is specialized in internet-related services [...]",
    "profile_image_url": "http://public.crunchbase.com/...",
    "domain": "google.com/",
    "homepage_url": "http://www.google.com/",
    "facebook_url": "https://www.facebook.com/Google",
    "twitter_url": "https://twitter.com/google",
    "linkedin_url": "http://www.linkedin.com/company/google",
    "city_name": "Mountain View",
    "region_name": "California",
    "country_code": "United States"
  }
}
```

Listing 6: Excerpt of Crunchbase ODM API result for "google.com," used to look up the associated organisation and additional properties like a profile image, short description, and country of origin.

Tracker Lists and Organisation Properties

The final organisation source used by IoT Guard is a combination of several tracker lists, which not only provide a fine-grained domain-to-organisation mapping but also various additional organisation properties like categories, short descriptions, and privacy policy URLs. However, there were several challenges in using such a combination of lists. One of the biggest was that these lists often contain different spellings or aliases for the same organisation, which prevents IoT Guard from merging their associated properties. To mitigate this issue, a manually curated alias list based on detected merge conflicts is used to support the existing normalisation mechanism (see Section 6.6.2). Another challenge was that different lists return different granularities for the same domain, like `YouTube` and `Google` for `youtube.com`, resulting in a loss of information when merging lists. While this is usually prevented through predefined granularities for each source (see Section 6.6.1), a single tracker list can vary in granularity within itself, requiring the additional use of a predefined subsidiary list, thus preventing fine-grained child organisations from being overwritten by their coarser parent organisations.

The following list provides a brief overview of the tracker lists that were used during the evaluation of IoT Guard, with their order representing the order they were merged, putting the – in the author's opinion – most reliable lists last, so that they override any unresolved conflicts with higher confidence:

- **X-Ray**: A small tracker list used in the research of Van Kleek et al., which is not updated any more, but still contains some relevant entries, especially about mobile apps [158].
- **WhoTracks.Me**: A regularly updated tracker list and active open research project from Cliqz, the owner of the popular Ghostery privacy browser extension, containing parts of their proprietary tracker database, privacy policy URLs, and tracker categories like "CDN," "customer interaction," or "advertising" [80].
- **Disconnect Tracking Protection**: An actively maintained tracker list used by the Firefox Enhanced Tracking Protection feature, which similarly groups trackers into categories like "advertising," "analytics," "social media," or "cryptomining" [31].
- **webXray**: Similar to *X-Ray*, the webXray tracker list is also used in academic research, and contains, among other properties, subsidiary information, aliases, privacy policies, and countries of origin. The list is, however, only updated irregularly [90].
- **Better Blocker**: A tracker list maintained by the Irish non-profit "Small Technology Foundation" (formerly Ind.ie), which is used in their open-source macOS and iOS tracker blocker. Their tracker database is available as a list of semi-structured Markdown files containing categories, subsidiary information, short descriptions, and some additional notes, which include, for example, relevant excerpts of the organisation's privacy policy [147].

- **Exodus Privacy**: A tracker list focused on mobile apps that is maintained by the French non-profit organisation with the same name. As the list contains extensive descriptions for some trackers, IoT Guard tries to extract excerpts, and, similar to the *WhoTracks.Me* and *Better* lists, adds a link to the full tracker profile on the project website [41].

6.6.5 Additional Organisation Properties

As already mentioned for Crunchbase, IoT Guard also includes several resources that provide additional organisation properties given their name, in particular for organisations collecting audience-related data, as briefly discussed below:

- **TrustArc Preference Manager**: A compliance service providing a short description, categories, and a link to the privacy policies of advertising-related organisations [154], similar to the Digital Advertising Alliance (DAA) WebChoices service [30], which were both scraped by IoT Guard for offline access.

- **Evidon organisation profiles**: Evidon, the former parent company of Ghostery, offers another compliance service by providing profiles of marketing-related organisations that include various privacy-related properties like the type of data collected and its intended use [40], as seen in Figure 6.3.

- **Global Vendor List**: A list of advertising-related organisations and their data collection purposes, maintained by the Interactive Advertising Bureau (IAB) to support organisations in their GDPR compliance. Version two of the Transparency & Consent Framework (TCF) was released during the write-up of this thesis, and contains more precise purposes like "Create a personalised ads profile," "Develop and improve products," and "Use precise geolocation data" [68]. IoT Guard uses these purposes to highlight potentially unwanted data collection to the user, as shown in Section 7.3.4.

- **AS Organisations Dataset**: A dataset collected by the Center for Applied Internet Data Analysis (CAIDA) that is used to augment the AS organisations from the MaxMind ASN database (see Section 6.6.3) with their respective countries of origin [17].

Similar to organisation names, IoT Guard also performs normalisations of organisation properties in order to reduce the number of slight variations. For example, by combining the categories "site analytics," "analytics provider," and "analytics" into "attribution / analytics." Similarly, country of origin properties consisting of only an ISO 3166 alpha-2 country code are normalised to their corresponding display names, like "AT" to "Austria" [77].

Website: https://www.appnexus.com/en

About Us: https://www.appnexus.com/en/company/about-us

In Their Own Words

AppNexus is a technology company whose cloud-based software platform powers and optimizes the programmatic sale and purchase of digital advertising. Because our platform is open, customers are able to build and differentiate their technology on top of ours, making AppNexus the infrastructure through which a substantial portion of the world's internet advertising flows. As an independent, pure-technology provider, we are fully aligned with our customers' interests. Our core mission is to help our customers deliver the right ads at the right time to the right audience.

Industry affiliations

What does this company do?

Ad Server
Technology that delivers advertisements to websites and monitors progress and performance of ad campaigns.

Data Management Platform
Provider of technology to manage collection, storage, protection, segmentation and use of online data. DMPs are also used to manage users' privacy preferences.

Demand Side Platform
Technology provider that enables advertisers to buy ad inventory from multiple ad exchanges utilizing multiple data suppliers and auction-based bidding.

Mobile
Provider of advertising services to wireless devices (phones, tablets, etc.), typically through mobile web browsers or mobile applications.

Supply Side Platform
Technology provider that helps publishers sell ad space through multiple networks and/or exchanges and optimize ad revenue.

What data does this company collect?

Data Collected
Anonymous: Ad Views, Analytics, Browser Information, Cookie Data , Date/Time, Demographic Data, Hardware/Software Type, Interaction Data , Page Views , Serving Domains

Pseudonymous: IP Address, Location Based Data, Clickstream Data, Device ID

PII: PII Collected via 3rd Parties, EU- IP Address, EU- Unique Device ID

*Data is collected on behalf of, and owned by, client.

Data Sharing
Aggregate data is shared with 3rd parties., Anonymous data is shared with 3rd parties., PII data is shared with 3rd parties.

Data Retention
4-5 Years

Data Use
Ad Serving, Ad Targeting, Analytics/Measurement, Content Customization, Optimization, Cross Device Tracking

Your choices

We believe this company facilitates or engages in 3rd party interest-based targeting.

Click here to opt out of AppNexus »

Learn about the Ghostery browser extension that can help protect your privacy settings.

Privacy contact

AppNexus, Inc.

28 West 23rd Street
4th Floor
New York, NY, 10010

Phone: (646) 723-7844
Email: privacy@appnexus.com
Privacy contact URL:
https://www.appnexus.com/en/company/privacy-form

Privacy policy

Privacy policy: https://www.appnexus.com/en/company/platform-privacy-policy

Opt-out cookies

When you opt out of being tracked and/or targeted by this company, an "opt-out cookie" is set in your web browser. The details of this cookie are below. Click here for more information about what this means. *

Browser cookies

Name: uuid2
Path: /
Content: -1
Expiration date: 1969-12-31 19:00:00

Name: _mkto_trk
Path: /
Content: id:204-KZG-685&token:_mch-appnexus.com-1519717108363-82361
Expiration date: 1982-04-05 19:34:08

Flash cookies/LSOs

No LSO usage.

Figure 6.3: Organisation profile for "AppNexus" provided by Evidon [40], which are scraped and analysed by IoT Guard to provide privacy-related properties about organisations.

6.7 Block Unwanted Organisations

With the higher abstraction of contacted organisations, IoT Guard is able to provide blocking controls for unwanted communication in a more user-friendly way, as desired by participants in Section 4.5, by dynamically populating a DNS blocklist based on the already discussed domain-to-organisation mapping. Using its heuristic organisation hierarchies, IoT Guard can further automatically block any identified child organisations. For example, blocking Facebook would also block any communication related to Instagram or WhatsApp. However, as this might be too restrictive for some users, IoT Guard also offers the option to whitelist entire subsidiaries or just individual Flows, as demonstrated in Section 7.4.

To realise such blocking, IoT Guard uses its MITM position (see Section 6.2) to deploy the open-source software *dnsmasq* as a transparent DNS proxy that selectively answers DNS requests for blocked domains with the non-routable all-zero IP address, effectively blocking the communication attempt [32, 65, 91]. Listing 7 shows an example of the dynamically generated blocklist that blocks `google.com` and all its subdomains, except the whitelisted subdomain `play.google.com` by forwarding such requests to the upstream DNS server, as indicated by the number sign (`#`) [82].

```
address=/google.com/0.0.0.0
server=/play.google.com/#
```

Listing 7: Dnsmasq blocklist showing a blocked domain and whitelisted subdomain, which is automatically generated by IoT Guard on user demand.

In comparison to predefined organisation blocklists, IoT Guard's dynamic approach is able to block new domains and organisations that are not yet known to these lists, but have been dynamically identified by IoT Guard's various sources. A disadvantage, however, is that this only works reactively, i.e., new organisations can only be blocked after a corresponding connection has already been established by a device. Furthermore, the use of DNS blocking may prevent changes from taking effect immediately, as devices might still have cached previous DNS responses [103]. However, these limitations are acceptable, because the main objective of IoT Guard is to provide usable transparency and control rather than a high security solution comparable to a firewall.

6.8 Device Properties

As a final task, IoT Guard also extracts properties about the devices themselves by analysing two additional application layer protocols and devices' MAC addresses to allow users to quickly identify their devices.

6.8.1 Organizationally Unique Identifier (OUI)

A simple way to extract a device's manufacturer is to look at the OUI of its MAC address, consisting of the first three octets. For example, `00:0E:58` identifies a *Sonos* device, as assigned by the Institute of Electrical and Electronics Engineers (IEEE) Registration Authority [74]. To simplify this lookup, IoT Guard reuses the regularly updated *Wireshark* manufacturer database [164].

6.8.2 Dynamic Host Configuration Protocol (DHCP)

To provide user-friendly display names, IoT Guard analyses DHCP `REQUEST` messages, in particular their *Host Name* option, which announces a device's name to the DHCP server, which is responsible for dynamically assigning IP addresses and other network-specific configuration parameters to devices in the network [3, 34].

6.8.3 Simple Service Discovery Protocol (SSDP)

Finally, IoT Guard also analyses SSDP discovery messages used for the presence management of Universal Plug and Play (UPnP) devices in the local network. Since SSDP is based on HTTP messages sent over UDP, parts of the HTTP analyser from Section 6.5.1 could be reused for their analysis. To discover devices, a `M-SEARCH` HTTP message is regularly sent to the well-known multicast address `239.255.255.250`, to which supported devices respond with a message containing a `LOCATION` header that provides a link to the device's UPnP device description [156]. This description, hosted by the device itself, is subsequently requested by IoT Guard, with Listing 8 showing a shortened description for a Philips Hue bridge, where IoT Guard extracts the device's name, manufacturer, and further model-specific properties.

6.9 Implementation and API Details

To conclude this chapter, this section gives a short overview of IoT Guard's implementation and its API, which is used by the web-based user interface, but might also enable other researchers to provide different visualisations and further analysis. To begin with, IoT Guard was implemented in the Java programming language using Java Native Access (JNA) to interact with the native libpcap library [152], which is also the sole external dependency, greatly simplifying its deployment in other environments. The web-based Angular user interface was scaffolded using JHipster [78], and interacts with the API via JSON. To give an example of the exchanged information, Listing 9 shows a shortened JSON response for a device's communication behaviour in the specified interval. Alternatively, the parameter `minutes=X` can be used to quickly return a device's activity in the last X minutes, with the custom path `sinceLastChecked` returning only device behaviour since the details were last requested (see Section 7.1.1). Finally, without any parameters specified, IoT Guard returns a device's activity since it was first seen.

```
<?xml version="1.0"?>
<root xmlns="urn:schemas-upnp-org:device-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <device>
    <deviceType>urn:schemas-upnp-org:device:Basic:1</deviceType>
    <friendlyName>Philips hue (192.168.0.131)</friendlyName>
    <manufacturer>Signify</manufacturer>
    <manufacturerURL>http://www.meethue.com</manufacturerURL>
    <modelDescription>Philips hue Wireless Lighting</modelDescription>
    <modelName>Philips hue bridge 2015</modelName>
    <UDN>uuid:UUID</UDN>
  </device>
</root>
```

Listing 8: Shortened UPnP device description of a Philips Hue bridge, providing several details like its name and manufacturer, allowing users to quickly identify their devices.

```
GET /api/devices/<ID>?start=2020-02-17T21:12:00Z
                      &end=2020-02-17T21:23:00Z
```


```
{
  "id": "<ID>",
  "mac": "<MAC>",
  "firstSeen": "2020-02-16T20:50:15.177207Z",
  "lastSeen": "2020-02-17T21:23:00Z",
  "newSince": "2020-02-17T21:12:00Z",
  "uploadSeries": [0,39,0,39,6750,39,0,31260,0,39,0,39],
  "downloadSeries": [0,35,0,35,3505,35,0,10146727,0,35,0,35],
  "chronoUnit": "MINUTES",
  "chronoStep": 1,
  "uploadTotal": 38205,
  "downloadTotal": 10150407,
  "contactedOrganisations": 3,
  "potentiallyUnwantedOrganisations": 0,
  "properties": {
    "HOST_NAME": [
      {
        "type": "HOST_NAME",
        "value": "Philips hue (192.168.0.131)",
        "reliability": "HIGH"
      },
      {
        "type": "HOST_NAME",
        "value": "Philips-hue",
        "reliability": "MEDIUM"
      }
    ]
  },
  "flowTree": {"organisation":{"name": "Philips", ...}}
}
```


Listing 9: JSON response excerpt for a device's activity in the requested interval, containing the overall time series for the device and a selection of its properties, demonstrating how IoT Guard deals with multiple property sources through reliabilities. Listing 10 continues with the "flowTree" part of the response, containing the contacted organisations and Flows.

```
{
  "organisation": {
    "name": "Philips",
    "uploadSeries": [0,39,0,39,6750,39,0,31260,0,39,0,39],
    "downloadSeries": [0,35,0,35,3505,35,0,10146727,0,35,0,35],
    "uploadTotal": 36673,
    "downloadTotal": 10146240,
    "newTree": false,
    "anyNew": true,
    "blocked": false,
    "potentiallyUnwanted": false,
    "properties": {
      "COUNTRY": [
        {
          "type": "COUNTRY",
          "value": "Netherlands",
          "source": "Whois",
          "potentiallyUnwanted": false
        }
      ],
      "DESCRIPTION": [
        {
          "type": "DESCRIPTION",
          "value": "Koninklijke Philips is a technology ...",
          "source": "Crunchbase",
          "potentiallyUnwanted": false
        }
      ]
    }
  },
  "children": [{"organisation": {"name": "Philips Hue", ...}}],
  "flows": [{"id": "philips.com", ...}],
}
```


Listing 10: The flowTree part of the JSON response from Listing 9, containing the contacted organisations, their properties, and the corresponding Flows in a hierarchical structure. The Flow details are continued in Listing 11.

```
{
  "id": "philips.com",
  "organisationName": "Philips",
  "uploadTotal": 15846,
  "downloadTotal": 10144756,
  "blocked": false,
  "new": false,
  "subFlows": [
    {
      "id": "fds.dc1.philips.com",
      "ports": [{"port": 80, "protocol": "TCP"}],
      "uploadTotal": 188,
      "downloadTotal": 10128085,
      "blocked": false,
      "new": true
    }
  ],
  "externalAddresses": ["54.72.223.127", "52.213.14.62", ...],
  "requests": [
    {
      "first": "2020-02-17T21:19:22.337093Z",
      "lastTime": "2020-02-17T21:19:00Z",
      "method": "GET",
      "host": "fds.dc1.philips.com",
      "target": "/firmware/.../product.RSA_prod_01.fw2",
      "count": 1,
      "new": true
    }
  ],
  "organisationGuesses": [
    {
      "organisationName": "Philips",
      "source": "Whois",
      "granularity": "ORGANISATION",
      "reliability": "MEDIUM_HIGH"
    },
  ]
}
```


Listing 11: The Flow details part of the JSON response in Listing 10, containing detailed information about the contacted domains, used ports, and extracted requests. Furthermore, it contains insight into the guesses that assigned the organisation to the Flow. Note: IoT Guard also records a time series for each Flow and sub Flow, but they are currently not exposed via the API for space reasons, as they are not used by the GUI.

CHAPTER 7

Visual Design and Interaction

The previous chapter showed how contacted organisations and additional information can be extracted from smart devices' network traffic and external resources. This chapter presents the visual encoding and interaction created by the author based on participants' requirements and sketches, without directly copying any of their designs. As already mentioned in Section 2.2.1, the design underwent several iterations based on the six informal usability tests performed with non-expert users. While this chapter focuses on the final design, it contains some insight into important changes to the visual abstractions based on participants' feedback. The following sections present the user interface according to the applied *Visual Information Seeking Mantra* [145]: overview first, zoom and filter, then details on demand; and concludes with the additional controls to block any unwanted communication.

7.1 Simple and Quick Overview

Participants' desire to quickly gain insight into a devices' communication behaviour inspired a majority of the visual abstractions used by IoT Guard, including the various highlights for new and potentially unwanted device behaviour, which also act as hints to investigate further details on demand.

7.1.1 Device Overview

To get an initial overview of active devices and their communication behaviour, IoT Guard provides a device overview (see Figure 7.1), which shows each active device as a card with the amount of transferred data visualised as a time series chart in the background, split by up- and download using red and green respectively. In addition to its name and manufacturer, each device card also contains the number of contacted organisations and how many of them are new and potentially unwanted, giving a preview

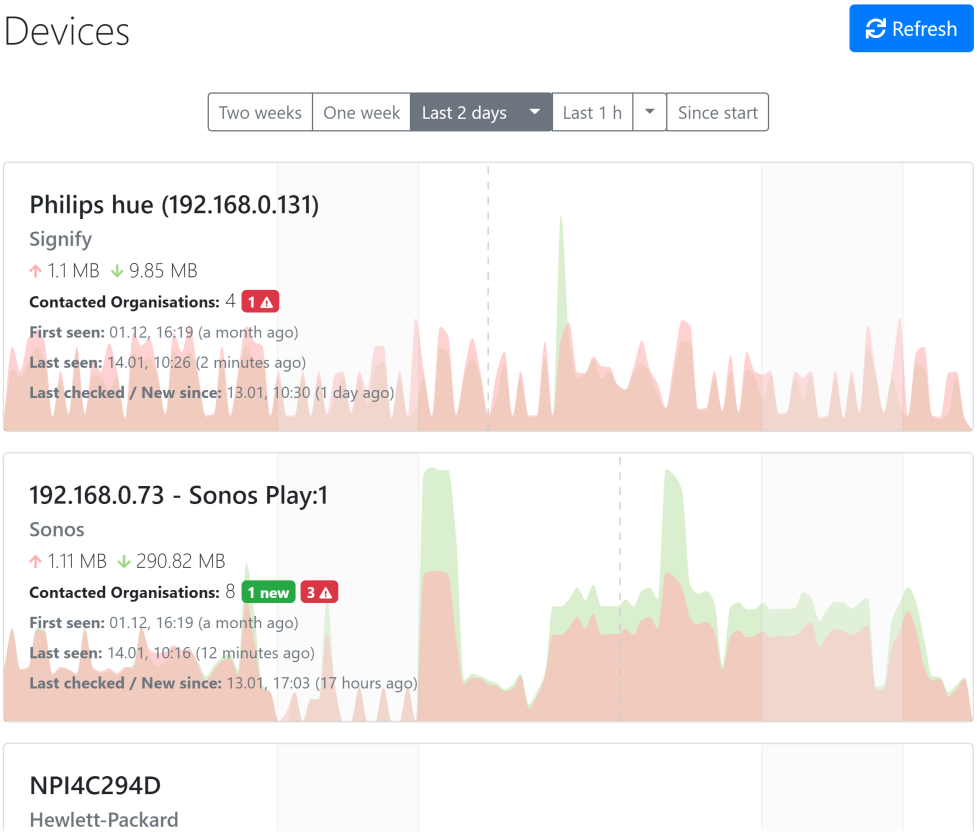


Figure 7.1: Device overview for the selected time span, giving a quick overview of the amount of transferred data and the number of contacted organisations. Clicking on a device card opens its details view as shown in Figure 7.2.

of its communication details so that users can quickly decide whether to investigate any further. New organisations are determined by the time the details view of a device was last checked, which is also indicated by the light grey dashed vertical line in the charts. To allow a comparison between devices, all charts are synchronised, and a logarithmic scale is used so as not to hide small but potentially unwanted data transfers.

Based on feedback by P9, all time series charts also contain nighttime highlights to quickly discover whether a device communicated even though it has most likely not been actively used, a concern already identified in Section 4.2.2. These highlights further improve the visualisation of multi-day time ranges, as they act as visual separators between the days.

7.1.2 Device Details

Clicking on a device card opens its details view (see Figure 7.2), which provides an overview of the contacted organisations for the selected interval, which is shown above the time series chart. Contacted organisations are visualised using cards that group all Flows

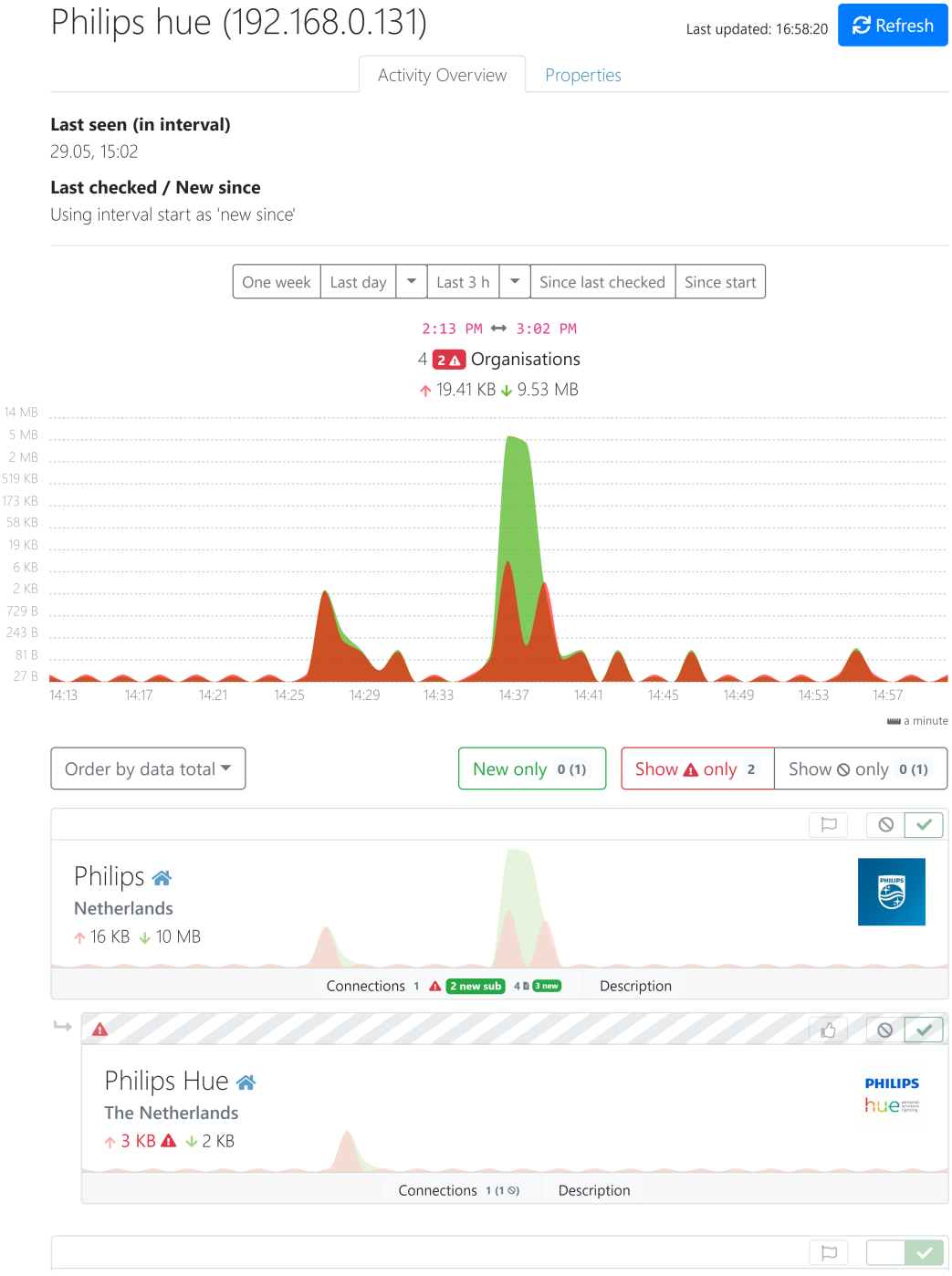


Figure 7.2: Device details for a Philips Hue bridge, zoomed in on a detected automatic firmware update, containing a hierarchical list of the contacted organisations as cards that provide further information on demand. The stripes on the "Philips Hue" card indicate that some of its Flows have been blocked.

assigned to that organisation, with their indentation and carriage return icon signalling a subsidiary hierarchy, whereas the icon was added based on user feedback. The additional organisation logos provided by the Crunchbase API (see Section 6.6.4) allow users to quickly identify familiar or potentially unknown organisations while scanning the list, as supported by the usability tests, with the house icon next to an organisation's name providing a quick link to its home page.

Similar to the device overview, the organisation cards use highlights and icons to indicate new and potentially unwanted information that can be further investigated on demand. The footer of an organisation card, for example, acts as a preview of its details tabs (see Section 7.3), as magnified in Figure 7.3, which highlights that the organisation contains one root Flow, with two new child Flows and some unencrypted communication, as indicated by the warning sign. Furthermore, it shows that four requests were extracted, whereas three of them are new.

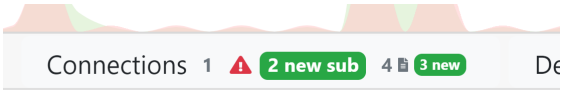


Figure 7.3: Magnified connection highlights as seen in the footer of the "Philips" organisation card from Figure 7.2, giving a preview of the details shown on demand.

Additionally, the subsidiary "Philips Hue" in Figure 7.2 features a red warning sign next to the amount of data to emphasise that it has uploaded (slightly) more data than it did download. Likewise, Figure 7.4 shows how organisation categories are used to highlight potentially unwanted communication behaviour like data collection for advertising-related purposes. Potentially unwanted countries of origin are similarly emphasised via red warning signs.

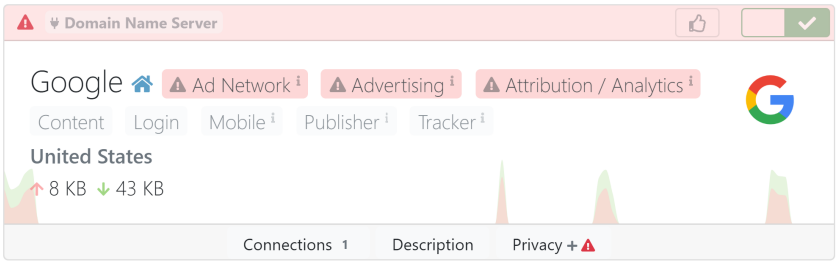


Figure 7.4: Protocol tag and organisation categories with warning highlights to give a quick indication of an organisation's purpose. The thumbs-up button can be used to mark an organisation as acceptable despite these warnings.

To support the discovery of such potentially unwanted behaviour, an additional warning sign is added in the left-hand corner of the card header. And based on user feedback, the header is further fully coloured red to improve its scannability. However, as participants desired to ignore warnings for organisations they deemed acceptable (see Section 4.5.5),

a thumbs-up button was added to the card headers that removes such highlights and warning signs, with an additional flag button to undo such decision or to manually mark an organisation as requiring further inspections.

To further improve the initial insight into why data has been exchanged with an organisation, IoT Guard adds a protocol tag to the card header if the organisation was contacted exclusively via this protocol, such as "Network Time Protocol," or "Domain Name Server," as seen in Figure 7.4. To avoid clutter, the most prevalent protocols HTTP and TLS are ignored for these tags.

Finally, in addition to the overall time series chart, each organisation card also contains a chart showing the distribution among the individual organisations and subsidiaries, whereas the chart of a parent again includes the sum of all its subsidiaries. Such detailed attribution of contacted organisations by time allows users to relate them to their interactions, which cannot be done automatically due to the technical limitation of analysing only the network traffic, as opposed to a direct analysis on the target device [79, 129, 149].

7.2 Zoom and Filter

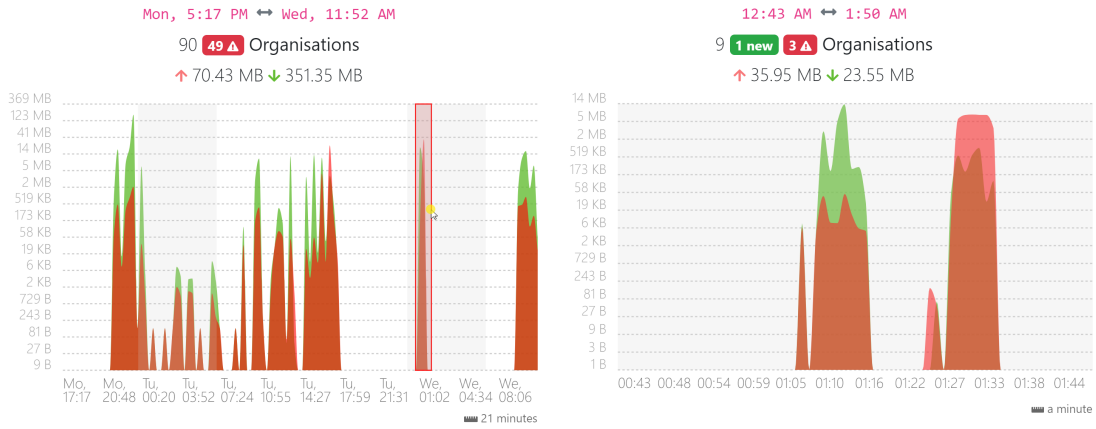
To aid the discovery of new and potentially unwanted communication, the user interface provides various ways to zoom in on areas of interest as well as filter and sort the contacted organisations.

7.2.1 Zoom in on Areas of Interest

An important design requirement was that IoT Guard should be able to analyse the behaviour of devices over a longer period in order to detect changes and enforce control. However, existing research has already cautioned against cluttering the visualisation with the growing amount of data [19]. IoT Guard therefore features several ways to influence the shown time interval and highlight changes. In the device overview, for example, the time range controls on top of the overview (see Figure 7.1) updates not only the devices' time series charts and their number of contacted organisations but also the active devices in that interval, sorting them by their last activity. To simplify the controls, a predefined selection of common ranges are provided that can be fine-tuned via drop down menus.

In the device details, the same time range controls are available, which update not only the contacted organisations and their charts but also their Flows, giving a detailed insight into the occurring communication during that interval. The additional "Since last checked" control (see Figure 7.2) further displays, as its name suggests, any device activity since the last time the user checked the device details. This timestamp also serves as the basis for highlighting new information.

Finally, for a more fine-grained range selection, the device chart itself allows users to zoom in on any interval of interest by dragging a zoom area with the mouse or with a



(a) Zoom start by dragging an area

(b) Zoom result

Figure 7.5: Zooming in on an area of interest in the device details chart, which not only updates the shown organisations and Flows but also highlights any new communication observed during that interval.

pinch gesture on touch devices, as demonstrated in Figure 7.5. Zooming in also updates the "new since" timestamp to the beginning of the chosen interval to emphasise the communication that was first observed during that interval, which may be used to investigate the cause of a spike in transmitted data.

7.2.2 Filter and Sort

In addition to updating the shown time interval, the user interface also provides controls to filter the view to only display organisations with new activity, organisations that are considered potentially unwanted, or organisations that were fully or partially blocked, as shown in Figure 7.6.

Figure 7.6: Magnified sort and filter controls for organisations from Figure 7.2, allowing users to further navigate the growing amount of data.

The filter buttons further preview the number of matching organisations and Flows, where "0 (1)" for the new filter means that although no organisation is new, one root Flow has some new activity. The new filter can also be combined with the other two, but the unwanted and blocked filter are mutually exclusive, as signalled through their toggle-inspired design, since blocked organisations are no longer considered unwanted. Selecting both the new and blocked filter allows users to quickly review and potentially whitelist any Flows or subsidiaries that were automatically blocked through the hierarchical blocking feature, as further discussed in Section 7.4.

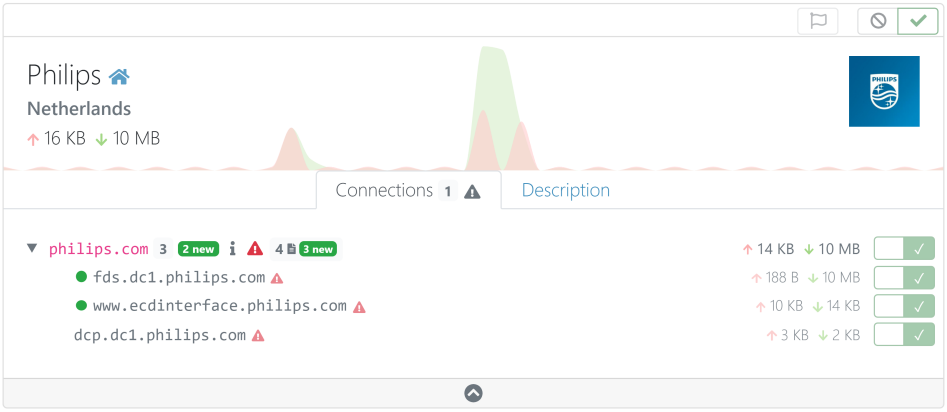


Figure 7.7: On-demand connection details for "Philips" from Figure 7.2, shown after clicking on the card footer. It contains the contacted Flows grouped by their root Flow, and similarly highlights new and potentially unwanted details.

Finally, the sort order of organisations and their Flows can also be changed through the dropdown shown in Figure 7.6, to sort by "data total," "data upload," "data download," "last activity," and "name," fine-tuning the displayed organisations.

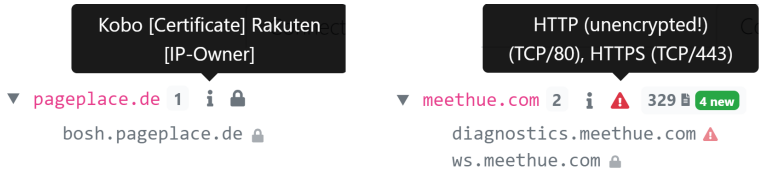
7.3 Details on Demand

Once an organisation of interest has been identified, its organisation card provides additional information in its details tabs, which can be opened on demand by clicking on its footer, containing the contacted domains, organisation descriptions, and privacy details, as discussed in detail in this section.

7.3.1 Connection Details

The first details tab contains the Flows assigned to the organisation (see Figure 7.7), grouped by their root Flow, and similar to the card footer, new and potentially unwanted information is highlighted in green and red respectively. In addition, each Flow shows the amount of transferred data in the selected interval, with the root Flow containing the sum of all child Flows.

Additionally, each root Flow contains further several details on demand. For example, the tooltip for the information icon (see Figure 7.8a) lists the Flow's organisation guesses sorted by granularity, starting with the finest and the one responsible for assigning the Flow to that organisation. The remaining guesses give an insight into the creation of the heuristic organisation hierarchy, and may increase confidence in the decision if several sources provide the same result, or otherwise show how conflicting results have been resolved, as desired by P5 in Section 4.6.2.



(a) Organisation guess details (b) Protocol details

Figure 7.8: Tooltips showing Flow details on demand, providing insight into the decision-making process of IoT Guard.

Figure 7.9: Extracted HTTP requests for “philips.com” from Figure 7.7, displayed as a popup after clicking on the document icon of the corresponding Flow. This example shows requests captured during an automatic firmware update.

Furthermore, the icon next to the guess details either shows a padlock symbol or a warning sign, depending on the application protocols used, with details similarly shown in a tooltip (see Figure 7.8b). Currently, IoT Guard only checks for HTTP to warn users about unencrypted communication. However, this might be extended to account for other potentially unwanted application protocols like Telnet or Secure Shell (SSH). The same protocol icons are also available for each child Flow to pinpoint exactly which Flow used which protocol.

Finally, a click on the additional document icon opens a popup containing the extracted HTTP requests that occurred during the selected interval (see Figure 7.9). To avoid clutter, the popup lists only unique request URIs and their frequency and sum of uploaded data, while sorting them by their last time and highlighting any new. For the same reason, their common root domain is replaced by the asterisk symbol, while their subdomain is italicised.

7.3.2 Inactive Connections

Although the overview is improved by displaying only Flows and organisations that were active in the selected interval, this may complicate blocking decisions, as the hierarchical blocking mechanism would also affect potentially hidden Flows and subsidiaries. To mitigate this issue, Flows and subsidiaries with no activity can be displayed on demand, with their availability signalled through crossed eye icons.

7.3.3 Organisation Descriptions

In the second details tab the general descriptions for the organisation are listed (see Figure 7.10), which include references to their sources that provide further details about the organisation and improve the attribution of the displayed properties.

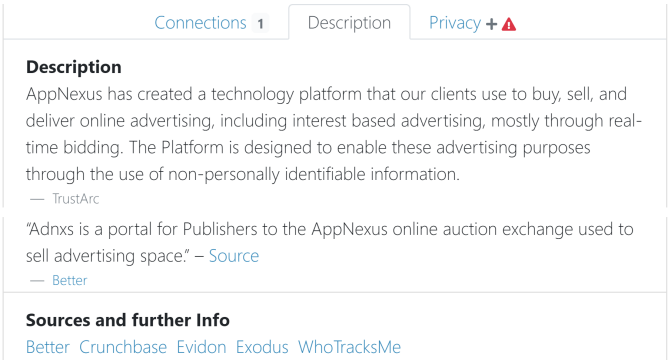


Figure 7.10: Shortened organisation descriptions for "AppNexus," with additional links to their sources, allowing users dig deeper if they wish.

7.3.4 Privacy Details

The third and final details tab summarises privacy-related properties of the organisation (see Figure 7.11). While a majority of sources include a link to the privacy policy, only the Evidon and IAB resources contain detailed information about the collected data and its use, with their availability indicated by a plus icon next to the tab name.

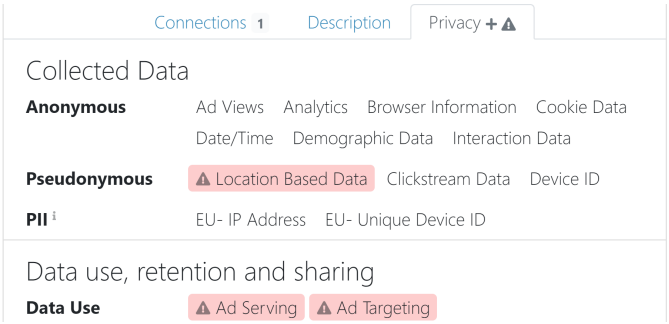


Figure 7.11: Shortened privacy details for "AppNexus" with highlights for potentially unwanted properties, and further descriptions on demand, as shown in Figure 7.12.

Similar to organisation categories, IoT Guard also highlights potentially unwanted privacy properties and indicates their presence via a warning sign on the tab header. The initial problematic properties were selected based on participants' preferences, which include, for example, location-based data and advertising-related data uses, that can be further customised through configuration files.

Finally, to improve the comprehension of data purposes like “Linking devices” and organisation categories like “Demand Side Platform,” IoT Guard also displays on-demand descriptions in popovers, with their availability indicated by small “i” icons next to the properties, as shown in Figure 7.12.

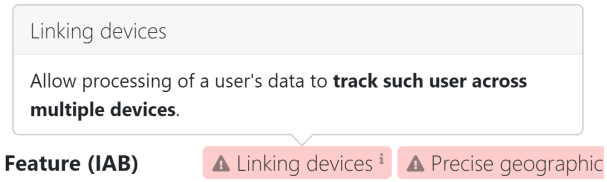
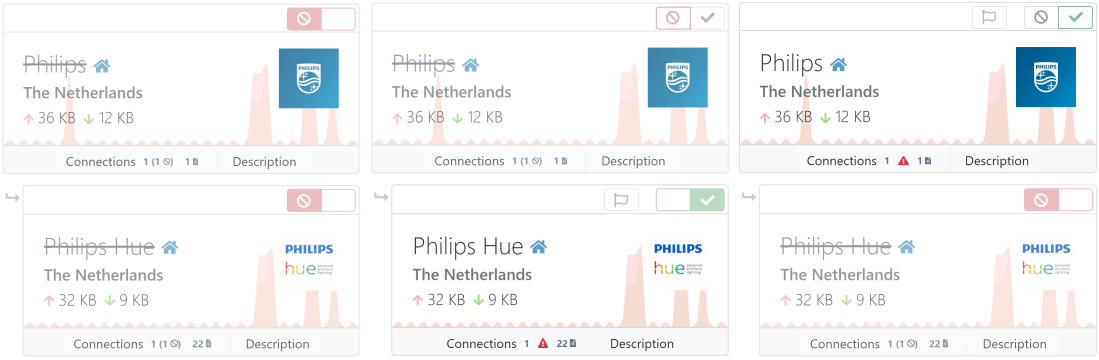


Figure 7.12: On-demand descriptions for organisations’ privacy properties

7.4 Block Unwanted Activity

If some unwanted communication behaviour has been detected, the user interface provides both coarse and fine-grained controls through the use of toggle buttons for whole organisations and individual Flows, which were inspired by the controls of P1 in Figure 5.6, and further refined based on user feedback from P6, P6w, and P3. Blocking a parent also blocks any current and future subsidiaries, with the possibility to whitelist individual subsidiaries, as shown in Figure 7.13, where blocked organisations are visualised by crossing out their names, removing any red potentially-unwanted highlights, and slightly reducing their opacity.

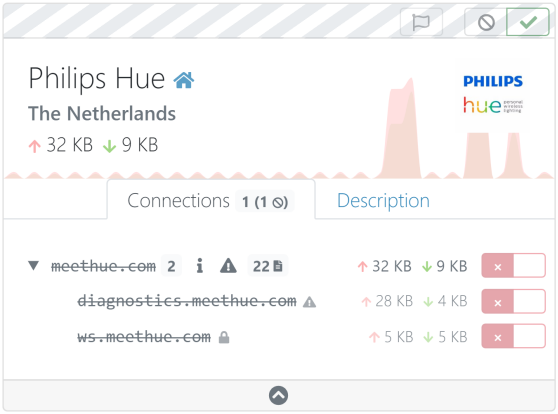


(a) Fully blocked (b) Whitelisted subsidiary (c) Just blocked subsidiary

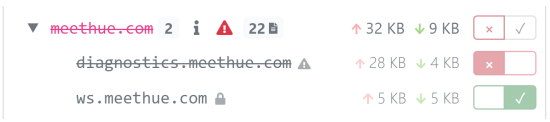
Figure 7.13: Different block states for an organisation hierarchy

A major design challenge for the block toggle buttons was that they should reflect not only the different blocking states for the current element but also for their child elements. The parent toggle button in Figure 7.13b, for example, shows both the block and check mark icon, with only the former partially coloured, to indicate that the organisation is blocked but also contains some whitelisted child elements. The reverse is shown in Figure 7.13c, where an accepted organisation contains some blocked child elements.

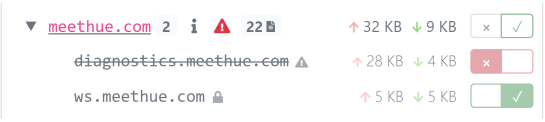
The same design elements also exist for blocked Flows, with additional highlights such as the striped organisation header in Figure 7.14a indicating some blocked Flows for an otherwise accepted organisations. Likewise, the accepted root Flow in Figure 7.14c is underlined to point out a blocked child Flow. In contrast, the blocked root Flow in Figure 7.14b is crossed out to indicate that any newly detected child Flows will be automatically blocked as well. This concludes the description of the visual encoding and interaction, with Chapter 8 concluding the design study by evaluating IoT Guard in real-world scenarios by target users and the author of this thesis.



(a) Fully blocked Flow hierarchy



(b) Blocked root, whitelisted child Flow



(c) Accepted root, blocked child Flow

Figure 7.14: Different block states for Flows. Note that the block states were changed for demonstration purposes of the visualisation only without waiting for changes in behaviour and therefore do not affect the amount of transferred data and number of extracted requests. Same for Figure 7.13.

CHAPTER 8

Evaluation and Discussion

The final task of this design study is to evaluate the created visualisation tool on real-world problems with target users in order to show that it actually provides users with transparency and control over their smart devices' communication behaviour. The literature identified several ways to validate the effectiveness of a chosen abstraction [76]. In this thesis informal usability tests with six non-expert users, and real-world deployments by three users and the author have been used to collect and investigate anecdotal evidence of IoT Guard's utility [110, 116], with the methodological details already discussed in Section 2.2. The following sections present a selection of the gained insight; summarise the qualitative user feedback and the identified limitations; and finally compare this thesis with related work, demonstrating how the new visualisation outperforms existing approaches.

8.1 Gained Insight

To start with, this section summarises the insight gained during the evaluation of IoT Guard in the informal usability tests and the real-world deployments, both using real home network traffic.

8.1.1 Unexpected Background Communication

In general, the evaluation showed that IoT Guard could successfully make users aware of potentially unknown and unexpected background communication of their devices. P9, for example, was really surprised when he saw that his iPhone had contacted Johannes Kepler University (JKU), the university where he studied several years ago, and immediately began checking his phone to see where it might have come from. It turned out that the Safari browser for iOS refreshed the favicons of his bookmarks, a behaviour already discovered as potentially unwanted by other researchers [36]. However,

Figure 8.1: Safari for iOS contacting a selection of bookmarked web pages on each application restart without any user interaction. In this example the author's long forgotten bookmark for his old maths exam. While the requests indicate that Safari is primarily refreshing the favicons of bookmarked websites, the fourth entry also revealed that Safari occasionally even fetches the full page.

further investigation by the author revealed that Safari occasionally also requests the actual bookmarked page in the background, as shown in the fourth request in Figure 8.1. Such behaviour could allow website owners to track users, since Safari regularly updates at least a selection of bookmarks on each app restart.

For P6 IoT Guard was able to detect unwanted communication to *King.com*, the maker of "Candy Crush Saga," on a fresh Windows 10 Home installation, even though he never interacted with the app or was aware of its existence. Likewise, IoT Guard also revealed that his smart TV was regularly communicating with Netflix, although he neither used the app nor has a Netflix account (see Figure 8.2), a behaviour which has been similarly observed in a recent analysis by Ren et al. [130]. In the same way, P6w and P9 were generally surprised by the large amount of communication that occurred on their phones while not interacting with them.

For P12, the deployment of IoT Guard in his home allowed him to discover that his BroadLink smart plug had some unwanted background communication with its Chinese manufacturer, even though he never used any of its remote control functionality. As a response, he not only blocked the communication in IoT Guard (see Figure 8.3) but also checked the device's configuration page to disable the unused remote control feature. Furthermore, P12 also discovered an unknown device from the manufacturer "Compal Information" in his home network that showed some communication to Google during the night, but did not follow up with details on how he investigated the device any further.

Finally, and as already mentioned multiple times in Chapter 7, the author of this thesis could also discover an unwanted diagnostics functionality of his Philips Hue bridge, with Figure 8.4 showing such an unwanted request. While IoT Guard did not analyse or display the transferred content, the author decided to block the responsible sub domain "diagnostics.meethue.com," based on encountered request parameters like "flash_lifetime," and "bridge_homekit_stats," in order not to potentially reveal private usage behaviour to the manufacturer.

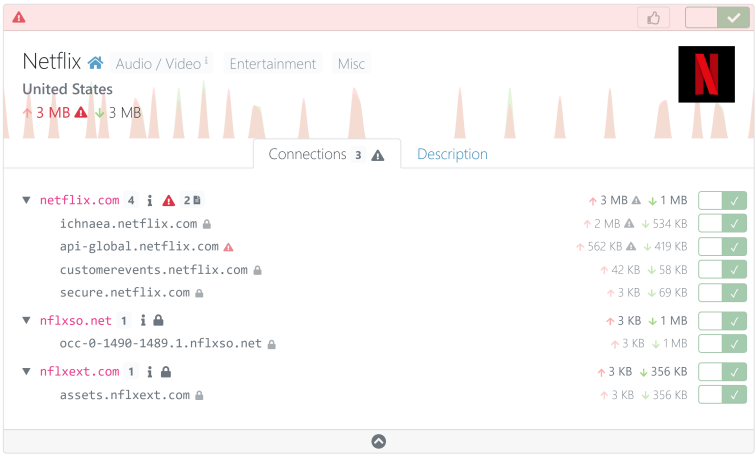


Figure 8.2: Unwanted background communication from a smart TV to Netflix over a span of two weeks, even though the app has never been used during this time or was ever launched before.

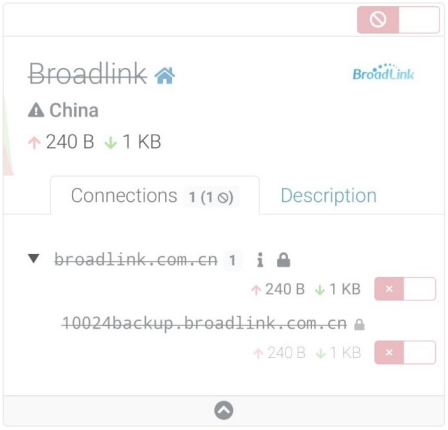


Figure 8.3: Blocked unwanted remote control functionality of P12's BroadLink smart plug. IoT Guard's visualisation further motivated him to disable said functionality in the device configuration.

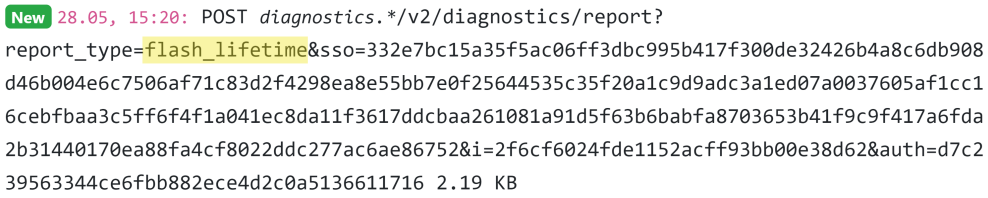


Figure 8.4: Unwanted diagnostics requests of a Philips Hue bridge to "diagnostics.meethue.com." Note the "report_type" parameter, which has been observed with values like "flash_lifetime," "bridge_homekit_stats," "ws_stats," etc., which could potentially reveal private usage behaviour to the manufacturer. In response, said diagnostics domain was blocked without noticeably affecting the functionality of the device.

8.1.2 Unwanted Tracking Behaviour

In addition to generally unexpected background communication, IoT Guard also revealed some unwanted tracking behaviour of devices and services.

Smart TV Tracking

The deployment of IoT Guard in the household of P6 and P6w, for example, revealed a previously unknown tracking functionality on their smart TV, which is related to the Hybrid broadcast broadband TV (HbbTV) standard used by TV channels to provide additional web-based content, which can usually be accessed by pressing the red button on the remote control [54]. While this functionality allows convenient access to, for example, Video on Demand (VOD) content of channels, the fact that these HbbTV applications are automatically loaded in the background without any user interaction raises some security and privacy concerns, especially since they can run arbitrary JS code, and were often loaded over insecure HTTP connections, as observed during the evaluation and by existing research [54].

To give an example of such HbbTV tracking application, Figure 8.5 shows how IoT Guard uncovered an audience measurement script on the Austrian ServusTV channel, which reports the presence of viewers once per second to a server operated by Red Bull Media House, the owner of ServusTV [143]. Some further investigations revealed that the measurement script was developed by Red Bull in cooperation with Fraunhofer FOKUS in order to provide live performance metrics of a broadcast [48]. While such statistics are useful for channel owners, the overly aggressive tracking behaviour poses several privacy challenges, in particular because further analysis of the script showed that it uniquely identifies a viewer's smart TV, and is therefore able to track reoccurring visits in addition to their watch time, potentially revealing private viewing habits.

Furthermore, opting out of this audience measurement requires a total of five button presses on the remote control, with the setting being placed on the last page of the privacy notice, making it difficult to discover without prior suspicion. More importantly,

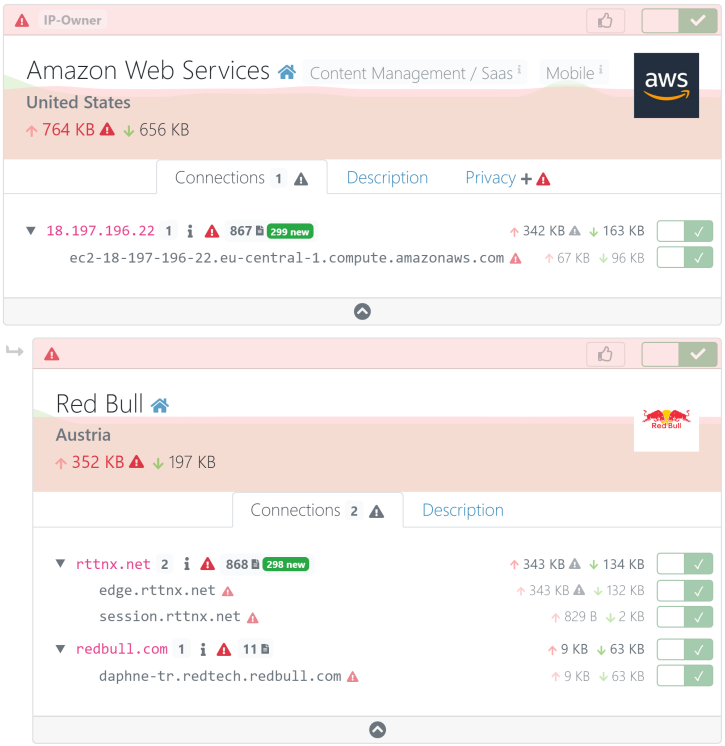


Figure 8.5: Detected tracking behaviour of the ServusTV HbbTV audience measurement script, showing the detailed tracking within only 14 minutes, which can potentially reveal private viewing habits.

opting out also does not prevent the initial script from being loaded in the background, which means that Red Bull is still able to track any user switching to their channel, with the ability to change or extend the tracking functionality at any time. A final review of their measurement script during the write-up of this thesis actually revealed that an additional tracking domain, also owned by Red Bull, was periodically contacted, which was not affected by the opt-out. This further underlines the need for such a tool, not only to detect such tracking behaviour but also to prevent the responsible HbbTV script from being loaded in the first place, by, for example, blocking a smart TV's access to Red Bull.

To mention another example, IoT Guard could also observe several HbbTV applications sending tracking data to third parties, with Figure 8.6 showing how the title and genre of a watched VOD broadcast was sent to the third party XiTi, an analytics suite owned by AT Internet[1].

[1] https://www.atinternet.com/en/

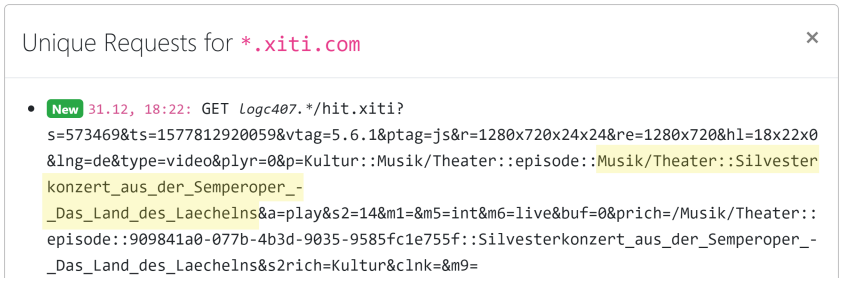


Figure 8.6: A detected leak of the broadcast title (highlights added) to the third party tracking service XiTi while using the HbbTV VOD service of the German channel ZDF.

Size of Tracking Landscape

Moreover, IoT Guard was also able to highlight to which extent users are tracked across the web, as demonstrated with a spontaneous, informal experiment during the usability test with P3, who considered herself generally as nonsceptical of tracking or targeted advertising, arguing that it "doesn't do her any harm." The task for her was to guess just how many organisations the popular tech news site The Verge[2] contacts when no adblockers or other tracker blockers like Ghostery are enabled, which she surprisingly had both installed despite her lack of concern. She guessed that roughly 20 more organisations would be contacted when visiting the site without tracking protection, having already seen how many have been contacted with the protection enabled. Due to the improvised experiment, the author likewise did not know the concrete number and guessed a total of 40 additional organisations. The results shocked both, because the total number of contacted organisations increased by 160 after reading only one news article on the site, with most organisations correctly flagged as potentially unwanted by IoT Guard. This large number even worried P3 and prompted her to ask the next day how she could block these organisations on her iPhone as well. This demonstrates IoT Guard's ability to raise at least initial awareness of the otherwise little known tracking landscape operating in the background of the Web. However, further research is required to evaluate any long-term changes in awareness.

8.2 Qualitative User Feedback and Limitations

After presenting the gained insight, this section summarises participants' feedback and problems during the informal usability tests and real-world deployments as well as general limitations of IoT Guard.

8.2.1 Positive Feedback and Observations

To begin with, the following list contains a brief summary of the informal feedback and observations gathered from participants' usage of IoT Guard:

[2] https://www.theverge.com/

- Overall, participants positively mentioned IoT Guard's ability to present such large amounts of data in a clear and concise manner without overwhelming them with clutter. P3 specifically emphasised the usefulness of organisation logos to quickly distinguish between known and unknown or potentially unwanted organisations, and likewise mentioned how the organisation hierarchy was able to show her that, for example, Instagram is owned by Facebook, with her being highly sceptical of the latter.

- The evaluation demonstrated that the final design of the highlights and warning icons could successfully communicate to all participants that some potentially unwanted behaviour was detected and further investigation is advised. Similarly, all participants discovered and explored the available details on demand without any instructions, and intuitively used the last design iteration of the blocking controls correctly.

- P12 particularly praised IoT Guard's plug-and-play capability, as it required zero configuration to start analysing his devices, and he could quickly uncover some unwanted communication behaviour of his smart plug, as already mentioned in Section 8.1.1.

- P9 highlighted IoT Guard's ability to block unwanted communication, which would allow him to buy, for example, a robot vacuum by the Chinese manufacturer Xiaomi in the future. As, according to his research, they are among the best on the market, but also feature a known tracking behaviour that he could now simply block.

- A further use for the blocking functionality was found by the author of this thesis by successfully preventing access to distractions like YouTube on all of his devices during labour-intensive periods, thus increasing his productivity. His colleagues also asked whether IoT Guard could schedule such blocks for specific periods of concentrated work, demonstrating the relevance of such a blocking functionality beyond privacy protection.

- Finally, P4 mentioned how such a tool could be used to detect unknown spying devices in, for example, AirBnB rental flats, a concern that has been proven to be true before [51].

8.2.2 Identified Problems and Limitations

In addition to praise, participants had a number of constructive criticism and suggestions for further improvements of the visualisation and IoT Guard in general, which are summarised with IoT Guard's overall limitations below:

- While the warning highlights were discovered by all participants, P6, for example, wished for more nuanced warnings that would distinguish between known malicious organisations and only potentially unwanted ones, as IoT Guard currently highlights

advertising-related organisation like Google and a known malware provider in the same way. He suggested the use of different colours or shades to represent the varying degrees of severity of the existing organisation categories.

- Related to warning highlights, P3 and P6 further mentioned that they would have liked more concrete hints and recommendations on how to proceed. P6 even went so far as to propose that IoT Guard could provide suggestions for disabling certain features of a device via its settings, or uninstalling an app from his smartphone. However, this would require more contextual information about the occurred communication, which is difficult to obtain by just analysing a devices' network traffic, as discussed in more detail below.

- Another issue with IoT Guard's warning highlights was that P3 incorrectly assumed that no highlights automatically meant that IoT Guard considered the communication acceptable, whereas it only meant that it could not detect any potentially unwanted behaviour. While IoT Guard also uses network-level properties such as excessive upload or unencrypted communication as indicators of potentially unwanted behaviour, the highlights of an organisation's purpose or its privacy properties ultimately depend on the quality and scope of the sources used.

  Similarly, the reliability of the domain to organisation assignment depends on the quality of the sources used, and, as already mentioned in Section 6.6, remains just an informed guess. Although IoT Guard uses a combination of multiple sources, most of which are updated automatically, and additionally resolves conflicts through a reliability property for each source, the result is still not authoritative, which should be better communicated to users appropriately.

- While IoT Guard was primarily designed for smart home devices, which often serve a specific purpose, participants also checked their smartphones' communication behaviour, with P4 and P6, for example, asking for more detailed information about which app caused the unwanted communication. However, this lack of attribution is inherent in a network traffic-based analysis and has already been mentioned as a drawback in existing research [130]. Nevertheless, IoT Guard's abilities to filter the displayed time span and highlight new information allowed the author to still successfully attribute unwanted communication to the app responsible, by using, for example, the "last 15 minutes" time filter after interacting with an app.

- P3 was further concerned whether organisations could detect if she had blocked any communication to them, and asked if it was even legal to do so. After receiving more details on the implementation of the blocking mechanism by relating it to a network-wide adblocker, and after being assured that the effect is easily reversible and not permanent, she became more interested in trying out IoT Guard's blocking functionality. This shows that more guidance and explanations on the inner workings of such a tool are needed to ensure that even non-experts can use it confidently. While this feedback has led to the addition of a FAQ page, further research is needed on how to effectively provide such guidance in place.

- While all participants successfully recognised the indentation of organisation cards as organisation hierarchies in the final design iteration, some were confused that it can also represent a service provider relationship (see Section 6.6.1). P4, for example, knew for certain that "Stadt Kärnten" was not a subsidiary of AWS, despite IoT Guard showing them in a hierarchy. Similarly, P3 wrongly assumed that, for example, the CDN Fastly was a parent organisation and not just a service provider.

  For this reason P4 proposed to completely hide service providers in the hierarchy. However, the author's evaluations have shown that they can provide valuable information in case not all domains of an organisation could be mapped correctly, but all used the same service provider and thus still remain visually close and connected. While IoT Guard tried to distinguish between those two types of hierarchies by adding "IP-Owner" badges to the left-hand side of service provider organisation cards (see AWS in Figure 8.5), these badges clearly failed to communicate their intended meaning, requiring further research on better visual abstractions.

- P12 also mentioned a current technical limitation of the blocking functionality in his feedback, as it currently applies to all devices analysed by IoT Guard, in contrast to the device-specific blocks he desired in Section 4.1. Future versions of IoT Guard could solve this issue by running multiple dnsmasq services for each device in parallel (see Section 6.7), or use an upcoming version of the dnsmasq fork FTLDNS of the Pi-hole project, which adds support for per-client blocking [63].

- Finally, it should be pointed out that IoT Guard currently does not persist any analysis results, and instead just keeps everything in memory. While this can be considered a desirable privacy feature, IoT Guard will eventually run out of memory. Future versions of IoT Guard could therefore automatically remove memory-consuming time series entries after a certain amount of time, or store them separately in a long-term storage.

In addition to these tool-related limitations, P9 also argued that such a tool would probably only be used if there was already some form of awareness of smart devices' privacy problems, further highlighting the problem with users' general lack of awareness, as discussed in Section 3.1.3. Moreover, none of the participants showed any interest in using IoT Guard over the long term, although this could be related to their general scepticism towards smart devices (see Section 2.1.1). This meant that they did not yet own potentially sensitive smart devices, which in turn could have limited their interest in monitoring the communication behaviour of their devices beyond their initial curiosity. However, more in-depth field studies with users owning a larger variety of smart devices are required to understand the adoption and use of such privacy tool. Nevertheless, the initial evaluation could already provide suggestive evidence that the chosen abstraction and visualisation can effectively and usefully convey smart devices' communication behaviour, providing a variety of opportunities for future research.

Finally, it should be noted that there are some future developments that might negatively affect the effectiveness of IoT Guard. One such development is DNS over HTTPS (DoH) [62], which, as its name suggests, uses HTTPS to resolve DNS queries and can therefore provide both integrity and confidentiality of the requested domains, but at the same time prevents IoT Guard from extracting these domains from DNS queries (see Section 6.5.3), and additionally limits its ability to block organisations (see Section 6.7). A similar development is the internet draft for encrypted SNI for TLS version 1.3 [133], which, similar to DoH, would prevent IoT Guard from extracting the contacted domain from the corresponding TLS option (see Section 6.5.2). While encrypted SNI is still in an experimental stage, Mozilla has recently rolled out DoH for all its US-based users [107]. While this limits IoT Guard's analysis capabilities, it can still extract the contacted domain from the SNI, and it remains to be seen whether DoH will be adopted by smart home devices in the near future as well. However, further research is recommended on how a tool can still provide users with usable transparency and control over their devices in such cases.

8.3 Related Work

The previous chapters have already shown that there exists a large body of research concerning privacy threats of smart home IoT devices, with the need for usable transparency and control already thoroughly discussed in Chapter 3. However, only little discussion exists on how to provide users with such transparency over their devices' behaviour. This section gives an overview of the existing approaches; what gaps were identified; how this thesis attempts to close them; and finally, in what ways the created visualisation improves the state of the art.

8.3.1 Smart Device Transparency

This section discusses selected research on transparency, starting with static privacy labels and continuing with dynamic smartphone and smart home specific work, highlighting the research gaps this thesis is meant to close. To begin with, Emami-Naeini et al. [37, 39] proposed the use of privacy labels, which summarise the most relevant privacy and security properties of a device, arguing that this could also solve the problem of little to no security and privacy-related information about devices at their time of purchase. The drawbacks of such a static label have already been discussed in Section 4.5.4.

A more dynamic approach was chosen, for example, by Razaghpanah et al. [129], who created an android application, which analysed the transferred data types to external domains contacted by users' apps. Several other smartphone-centric research had similar goals [118, 131, 149], but they all required TLS interception for their analysis, and most of their prototypes had to be run directly on the devices themselves, making them unsuitable in the smart home IoT context. In contrast, Chetty et al. [19] chose a more home-centric approach by analysing devices' communication behaviour from a central point in the network, but they primarily focused on the amount of data transferred by

each device, prompting participants to ask for more in-depth information in a more usable way.

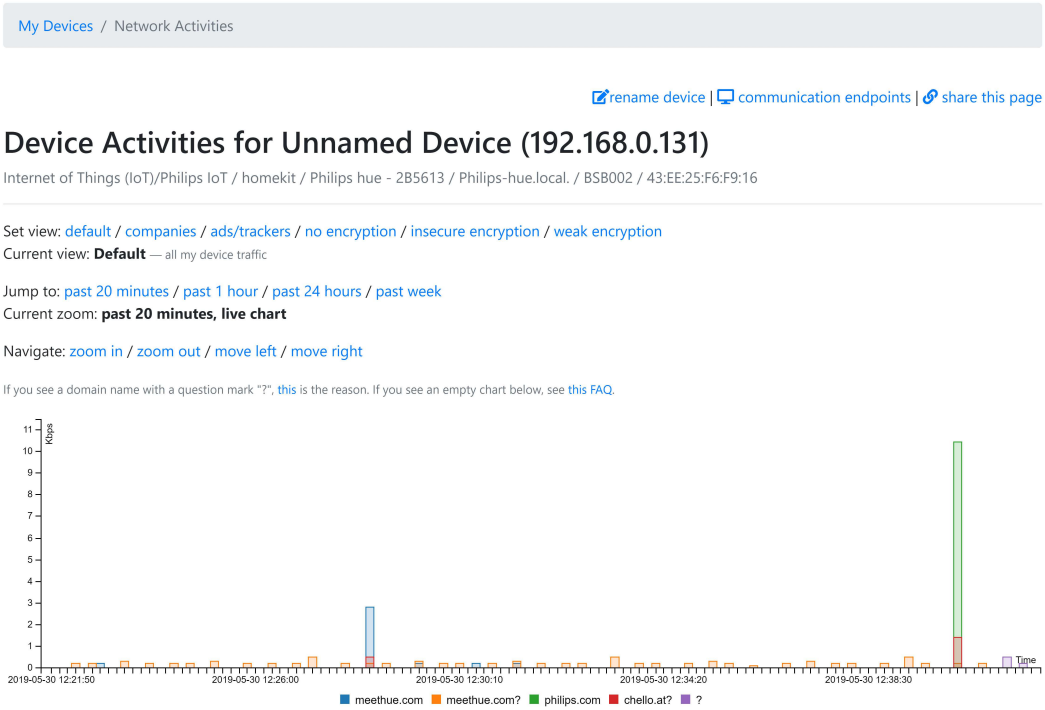
One recent attempt to provide more detailed information is the Princeton IoT Inspector by Huang et al. [67], which, similar to this thesis, analyses smart devices' communication behaviour over time and provides a visualisation of the contacted domains. However, their visualisation was only seen as a secondary goal of their research in order to keep users engaged in their field study, designed to collect smart devices' communication details for future research through crowdsourcing, and thus did not include users in the design process or evaluation of their tool. Additionally, due to their study design, the tool transferred and analysed the collected data on a remote server, and provided only a web-based user interface, which caused some privacy concerns among its users [126]. To give a comparison between the IoT Inspector and IoT Guard, Figure 8.7 and Figure 8.8 show the results of analysing the same Philips Hue device by both tools in parallel, demonstrating not only IoT Guard's more advanced organisation mapping with more details on demand but also its ability to block unwanted activity at different levels of granularity. This lack of control through the IoT Inspector was also criticised by its users [84].

Similar to this thesis, the recent research by Van Kleek et al. [158, 159] and Seymour et al. [144] focused more on users and identified their lack of awareness, lack of knowledge, and lack of control as three fundamental problems for smart home privacy. However, in their research they only included users through usability tests, without considering them in the design process (see Figure 8.9), which may have been the reason why participants occasionally got lost in their visualisation, and generally did not use the provided blocking functionality [144]. Nevertheless, the authors were able to show that their similar abstraction of contacted organisations not only allowed participants to reuse existing privacy preferences but also that it could successfully improve their awareness.

While the recent research by Ren et al. [130] did not focus on a visualisation of devices' communication behaviour, they analysed a wide range of smart devices in a lab setup, measuring their information exposure, based, among other things, on their contacted destinations, used encryption, and unexpected behaviour. Similar to this thesis, they used a combination of multiple sources to semi-automatically identify the contacted organisations from the extracted domain names, but did not provide any visualisation of their results. In contrast, IoT Guard provides similar and more information in a user-centric, real-time visualisation, with the contacted organisations determined fully automatically. Since Ren et al. published their data set[3], Figure 8.10 gives an example of how IoT Guard would visualise the unexpected data upload of a Ring doorbell in real time, as identified in their research [130].

Finally, the concurrent research by Yao et al. [169] similarly involved target users in a co-design study for smart home privacy mechanisms in order to elicit key design requirements and initial design ideas for non-expert privacy tools. This thesis supports

[3] https://moniotrlab.ccis.neu.edu/imc19/



(a) Bandwith chart

Remote Party	Country	Data Usage	Last Updated
? (35.190.23.141)	United States	470 Bytes	a few seconds ago
meethue.com? (35.190.23.141)	United States	7 KB	a minute ago
chello.at? (195.34.133.21 and 1 other IP address)	Netherlands	1 KB	a minute ago
philips.com	Ireland	7 KB	a minute ago
meethue.com	(unknown)	4 KB	10 minutes ago
google.com? Google (216.239.35.8)	United States	180 Bytes	22 minutes ago
(Local Network)	(unknown)	228 Bytes	23 minutes ago

Showing 1 to 7 of 7 entries Previous 1 Next

(b) Communication endpoints

Figure 8.7: Analysis results of the Princeton IoT Inspector for the same Philips Hue device as in Figure 8.8, showing how it failed to map the contacted domains to their organisations. Furthermore, their use of a non-logarithmic scale for the bandwidth chart is making less active domains increasingly difficult to spot.

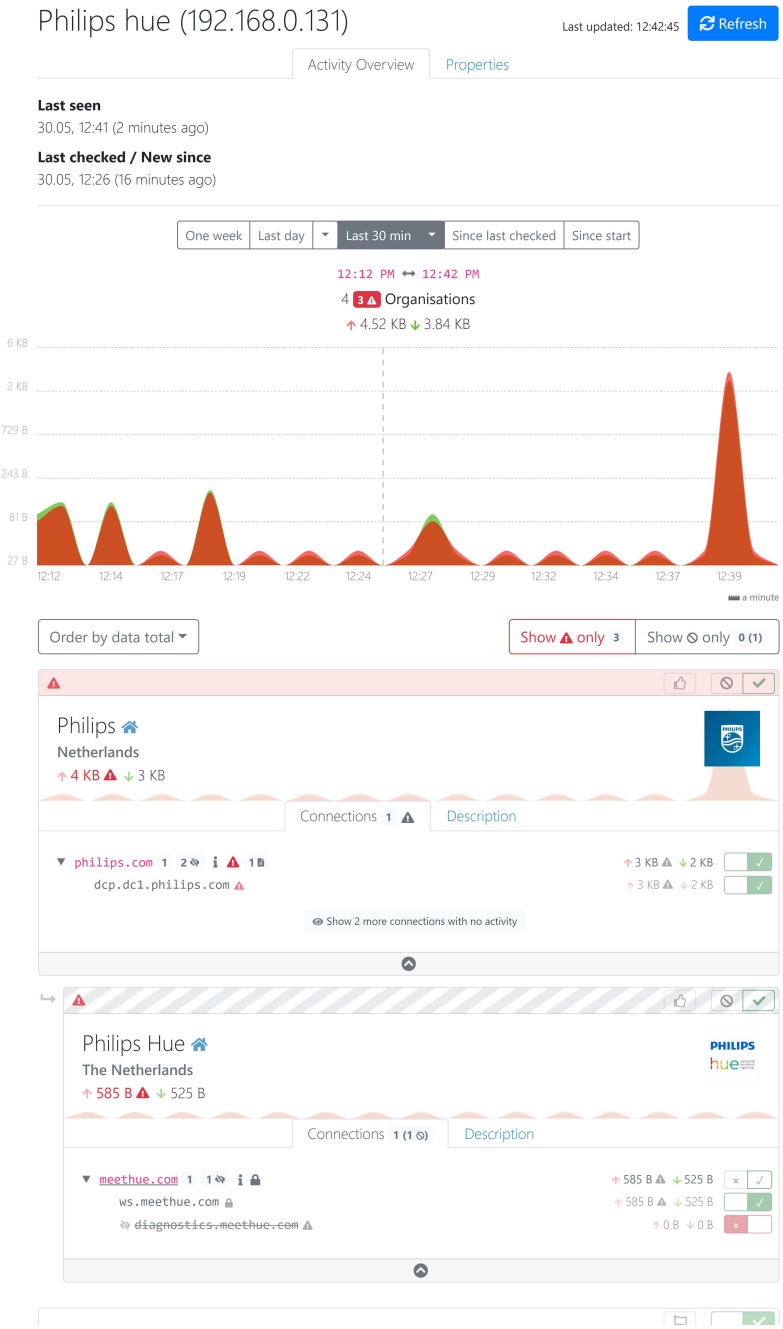


Figure 8.8: Analysis results of IoT Guard for the same Philips Hue device as in Figure 8.7, demonstrating its more advanced organisation mapping and its highlights for potentially unwanted behaviour. In addition, IoT Guard is able to display the contacted sub domains on demand, while providing fine-grained blocking controls, as demonstrated with the domain "diagnostics.meethue.com." (Note: The additional organisations "T-Mobile" and "Google" were cropped for space reasons.)

their qualitative results that transparency and control as well as usability are among the most important design requirements for such tools, and follows their call for future research on practical implementations with this user-centric design study, demonstrating one possible way how such transparency and control can be achieved. Concurrent to this thesis, Wilkinson et al. [163] investigated some design requirements for a glanceable data exposure visualisation, and while they focused on smartphone apps, they similarly concluded that such visualisation needs to include a simple overview, which highlights new information and potential threats, and provides further information on demand, a well-known concept already introduced by Ben Shneiderman [145].

8.3.2 Third Parties and Trackers

Various smart device-related research identified third parties and trackers as a major threat to users' privacy, arguing that the tracking ecosystem already known from the Web is equally relevant for these new class of devices [5]. This section provides a brief overview of related work on the analysis of trackers and third parties. One such ongoing research is the already mentioned "WhoTracks.Me" project by Karaj et al. [80], who, similar to this thesis, argued that contacted domains alone are not meaningful enough for users, and therefore created a mapping from domains to tracker organisations, further categorising them according to their purpose. This tracker database is used as one organisation source by IoT Guard. The newly released DuckDuckGo Tracker Radar follows a similar goal [35]. Furthermore, Razaghpanah et al. [128] studied third parties contacted by mobile applications, and emphasised the value of knowing an organisation's parent relationships, as their analysis of privacy policies revealed that they often allow data to be exchanged between them. This motivated the inclusion of the subsidiary hierarchies in IoT Guard.

A recent report by the Norwegian consumer protection association [45] further underlined through an in-depth analysis of popular and highly sensitive mobile applications just how extensive the tracking industry is collecting and sharing user data [20], and concluded that users have very little actual control over such data collection. Their report even led to three GDPR complaints being submitted to the corresponding data protection authority [113], highlighting once again the value of transparency over device- or app-related communication behaviour. Similar wide-spread tracking has been found in smart TVs and Over-The-Top (OTT) streaming devices in recent investigations by Varmaken et al. [160] and Moghaddam et al. [106], who both highlighted the incompleteness of existing block lists, and their tendency to break functionality. Merzdovnik et al. [101] similarly underlined the difficulty to block tracking in smart devices and recommended future research to support the laborious task of creating and maintaining such block lists. IoT Guard could be a first step towards easier block list creation, by providing a straightforward long-term analysis of devices, that can highlight any changes in behaviour. Furthermore, it can be used to directly prevent such unwanted behaviour of smart devices without requiring expert knowledge or a complicated setup.

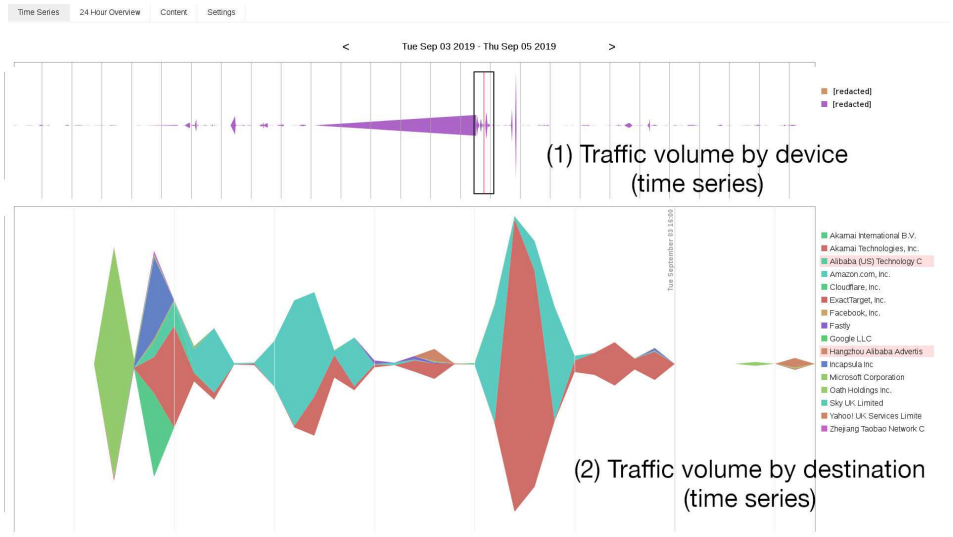
Webtracking Transparency

In addition to a scientific analysis of trackers, existing research also examined how tracker-related tools influence users' awareness and privacy concerns. Schaub et al. [140], for example, investigated users' reactions to three popular browser extensions in a qualitative lab study (see Figure 8.11), and concluded that although the extensions could raise participants' tracker awareness, they desired more information on the organisations' data collection purposes. Previous research has already tried to infer such purposes automatically, for example, through supervised machine learning [79], or by extracting and summarising relevant details from privacy policies [59, 86], as seen in Figure 8.12. Although IoT Guard does not automatically derive such information, it shows an aggregate of various organisation properties from existing sources.

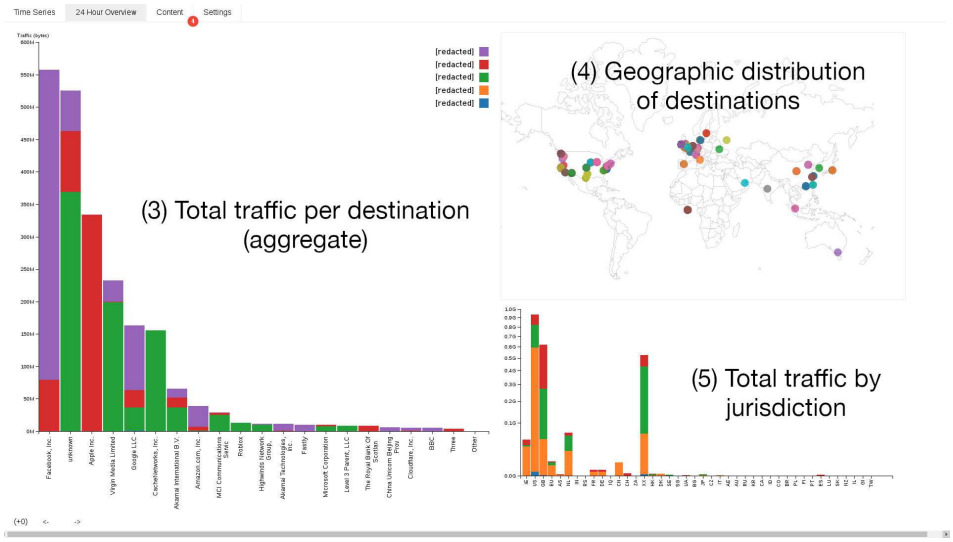
Similar research was performed by Weinshell et al. [162], who investigated how to better communicate the impact of tracking by showing users the interests that third parties might have derived from the websites on which they were found on (see Figure 8.13). Users were generally surprised by the large tracking landscape, with the authors concluding that such a detailed report could not only increase users understanding but also their desire to use more privacy preserving tools. As future work, they recommended additional research on tools that raise users' privacy awareness in order to provide a better basis for the discussion of trackers and PET. While IoT Guard does not provide such a personalised report on the impact of the observed tracking, it may serve as a starting point for future research on personalised privacy recommendations.

8.3.3 Other Transparency Research

A general introduction to Transparency Enhancing Tools (TETs) is given by Murman et al. [111] in their in-depth survey, in which they, among other things, categorised transparency-related user interface elements, summarised design guidelines and usability requirements from both a practical and legal standpoint, and similarly concluded that there is still a research gap for TETs in the IoT. Another research area related to this thesis is concerned with eco-feedback in households as provided by energy disaggregators, with Froehlich et al. [50], for example, investigating a visualisation of fixture-level water usage data, providing similar design recommendations like the ability to change the displayed time granularity. Similarly, Mennicken et al. [100] studied the visualisation of smart home configuration data and recommended highlighting potential anomalies, as participants were not interested in investigating the visualisation for each change themselves. Finally, Hamza et al. [58] examined how the new MUD standard [89] could be used to monitor the behaviour of smart devices in the home network, but didn't offer any details on a visualisation.



(a) Time series visualisation



(b) Aggregate visualisation

Figure 8.9: Visualisation provided by the Aretha design probe of Seymour et al. [144]. Their choice to use a bar chart to visualise the contacted organisations in (3) illustrates their problem of overwhelming the user and hiding small but potentially unwanted organisations. In addition, the chart shows that the second most frequently contacted organisation is "unknown," indicating problems with their organisation mapping, apart from a lack of further organisation details, requiring users to investigate each organisation's purpose on their own.

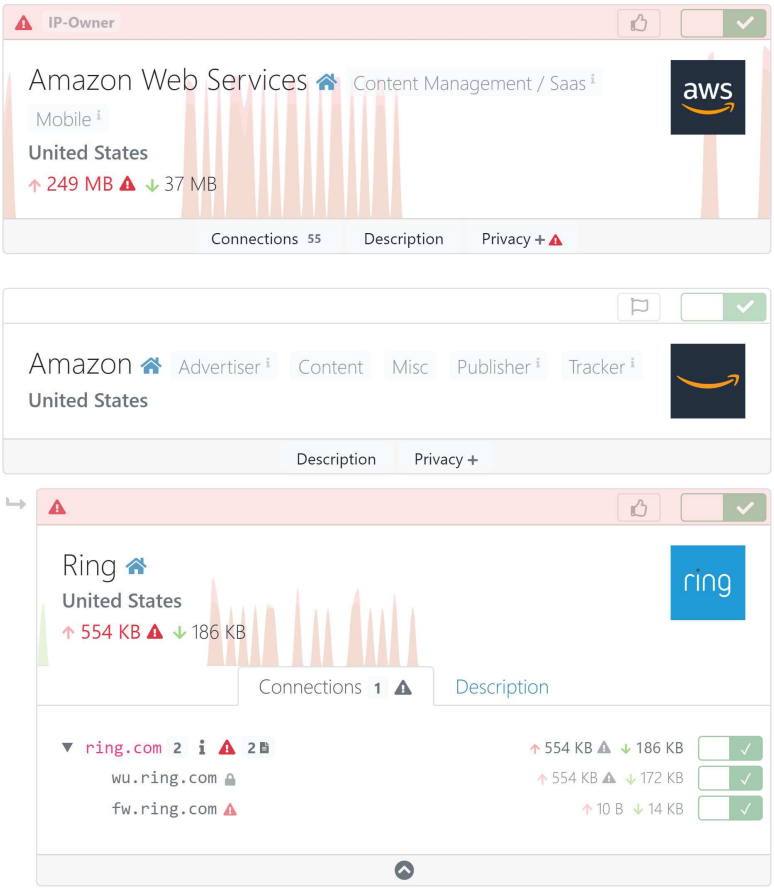


Figure 8.10: A visualisation of the unexpected data upload by the Ring doorbell from the data set of Ren et al. [130] using IoT Guard, demonstrating its ability to analyse previously captured device activity.

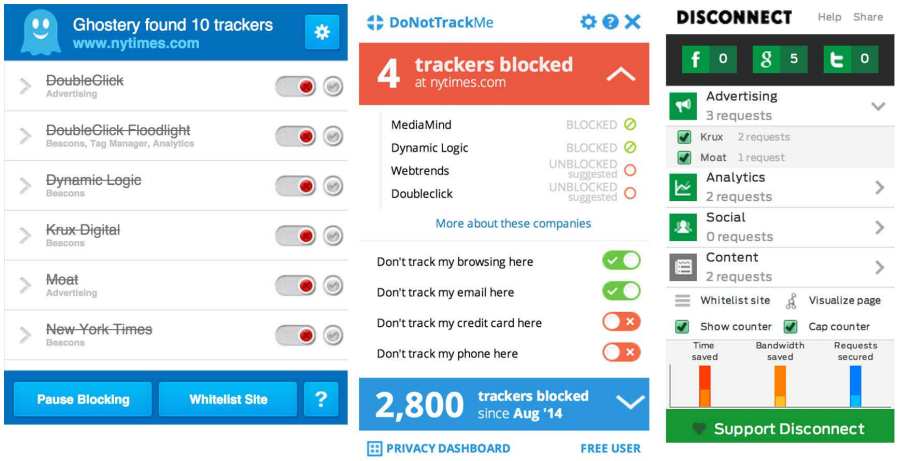


Figure 8.11: The tracker blocker extensions evaluated by Schaub et al., showing Ghostery, DoNotTrackMe, and Disconnect, with participants desiring more information about the individual organisations [140].

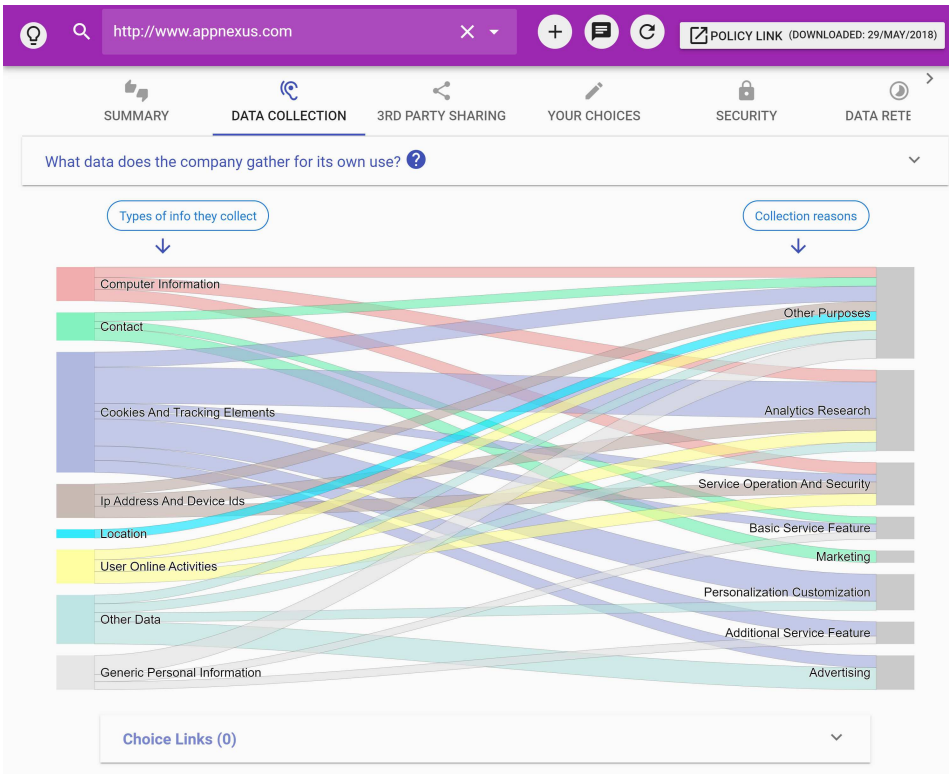


Figure 8.12: Example analysis result of the privacy policy of "AppNexus" using the Polisis website [59]. Hovering over one of the sankey arrows shows relevant excerpts of the privacy policy they extracted the information from. Future work could integrate these analysis results as an additional source in IoT Guard.

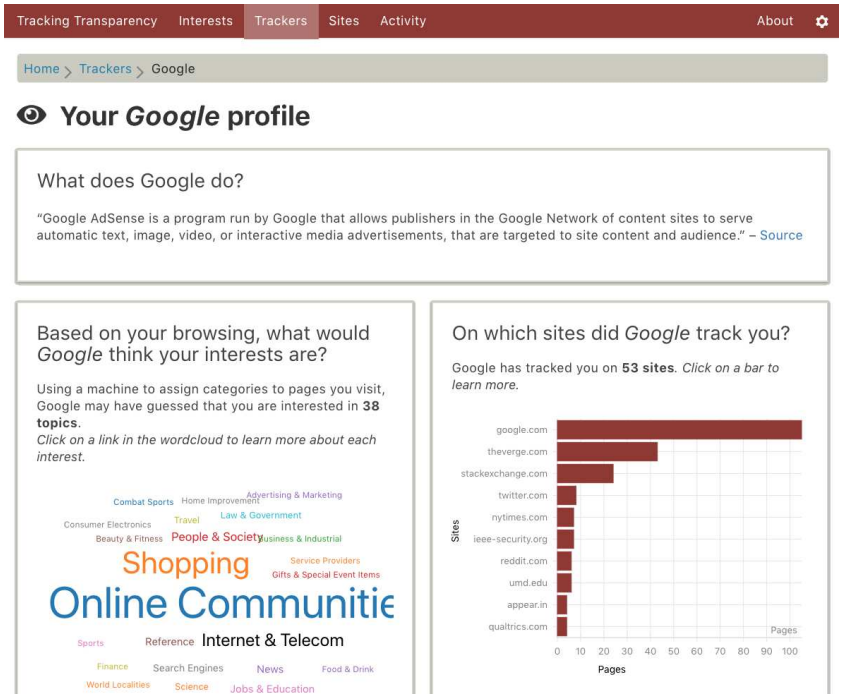


Figure 8.13: Example view of the Tracking Transparency extension by Weinshell et al., showing users what the selected tracker might have inferred from the websites the tracker was found on, which successfully increased participants' interest in privacy preserving tools [162].

CHAPTER 9

Conclusion and Recommendations

The goal of this thesis was to investigate how smart devices' outgoing network traffic can be visualised in a usable way to let users not only independently identify potentially unwanted communication behaviour but also make privacy-related decisions on their own. As summarised in Chapter 3, users often base their smart device-related privacy trade-offs on incomplete mental models and therefore risk privacy violations if their devices don't behave as expected. While related work has already shown that transparency can increase users' awareness, little work has been done on how to actually provide it in a usable way. This thesis therefore used a design study methodology to involve users directly in the design process through in-depth interviews, participatory design exercises, and informal usability tests in real-world environments, resulting in multiple design iterations of IoT Guard. Using a European and more sceptical demographic, this thesis could also strengthen existing qualitative results on the actual need for such transparency, and further extends the state of the art by describing how participants imagined a usable control over their devices, which has been identified as an essential requirement. Most surprising to the author, all participants wanted to decide for themselves what to block, based, among other things, on the purpose of the contacted organisation, which popular blocking tools like Pi-hole are not capable of. Nevertheless, participants still desired some guidance in the form of warnings and recommendations to support their decisions. The tool created in this thesis is meant as an initial exploration on how to provide such transparency and control to non-expert users, with the first evaluations offering suggestive evidence that it can effectively provide the desired insight with zero configuration overhead. To conclude this design study, the following sections summarise the identified and applied design guidelines, and discuss the possible future applications of such a tool.

9.1 Design Recommendations

The following design guidelines for usable transparency and control have been identified based on participants' inputs and were evaluated during the design study:

- Use contacted organisations as a primary abstraction for devices' outgoing communication behaviour, to allow users to reuse their existing mental models and privacy preferences. Furthermore, provide additional information about them, such as their purpose, country of origin, subsidiary relationship, and data processing practices, so users can assess unknown and potentially unwanted organisations without having to perform time-consuming research on their own.
- Start with a simple overview so that users can quickly see if their devices are behaving unexpectedly, while showing more details on demand. To provide such quick insight, highlight potentially unwanted behaviour with a combination of icons and red colour codings to quickly grab users' attention. However, users should be able to ignore these highlights to account for their individual privacy preferences.
- If users discover unwanted device behaviour, provide blocking controls that reuse the same organisation abstraction, so they don't have to manually investigate and block each connection. However, care must be taken to avoid breaking devices' overall smart functionality, which may be achieved through a fine-grained whitelisting ability.
- Visualise devices' communication behaviour over time and highlight any changes since the last time a user checked the tool, as initial results have shown that users are not interested in closely monitoring their devices over a longer period. Such a time-based visualisation may also highlight potentially unwanted communication patterns such as periodic diagnostic uploads or general device activity, even though the devices were not actively used. Similarly, tools should consider summaries such as monthly reports on device activity in order to reduce the effort required to monitor changes in behaviour.
- Finally, zoom and filter controls should help users investigate any behaviour of interest, and navigate the growing amount of data by focusing only on the most relevant information. Last but not least, users' trust in the provided information must also be considered, for example, by providing insight into the steps taken to derive the results.

9.2 Future Work

While a more in-depth evaluation is required to investigate IoT Guard's real-world adoption, these early results of its effectiveness already offer some insight into future research possibilities and further usage scenarios, which are briefly summarised below:

- As already mentioned, an important future research area is how such a tool is able to impact users' long-term privacy awareness, with P3's reaction to the large number of tracking organisations already providing initial results that IoT Guard is able to positively influence users' privacy awareness. In addition, further research is required to investigate how to provide better guidance and educational information on potential risks of third party data sharing, including concrete recommendations on how to reduce these risks.
- Similarly, more research is needed to investigate whether users actually trust such a tool to analyse their devices' communication behaviour (see Section 4.6), as the current evaluation only included participants already known to the researcher, which most likely positively influenced their trust.
- As already suggested by Hong [66], such transparency could also be used by journalists, policymakers, and app stores, in order to quickly uncover devices' or mobile apps' potentially unwanted communication behaviour and inform the public. More detailed and costly expert reviews would then only be required on demand [45]. This may also create better market incentives for manufacturers to improve the privacy of their products, as already mentioned in Section 3.1.3.
- Related to P6's desire for more nuanced highlights of potentially unwanted organisations, future work might also improve IoT Guard's blocking functionality by allowing users to block organisations by purpose or category.
- Future research may also explore whether such a tool could be used to create a crowd sourced organisation database, allowing users to add previously unknown organisations, or manually influence their categories and privacy properties, mitigating IoT Guard's dependence on external sources. This could also contribute to existing research projects like WhoTracks.Me [80] or the recently released DuckDuckGo Tracker Radar [35].
- Since IoT Guard has access to the full byte stream, further research might also perform more in-depth analysis of both encrypted and unencrypted communication in order to find potential privacy issues [130].
- Finally, because the Java-based tool requires no external dependencies other than libpcap, and provides a JSON interface for its analysis results (see Section 6.9), other researchers can easily create different real-time visualisations of devices' communication behaviour without having to re-implement the time-consuming low-level analysis from scratch.

In conclusion, this design study could not only strengthen the need for usable transparency and control of smart devices' communication behaviour but also make a timely contribution to the research field of usable privacy and security by identifying several design guidelines, and providing suggestive evidence of their effectiveness through their implementation in a tool that was evaluated with target users in real-world scenarios.

List of Figures

Acronyms

ARP Address Resolution Protocol. 62, 63

AS Autonomous System. 70, 75

ASN Autonomous System Number. 70, 75

AWS Amazon Web Services. 70, 103

CAIDA Center for Applied Internet Data Analysis. 75

CDN Content Delivery Network. 67, 73, 74, 103

CI Contextual Integrity. 2, 27, 28, 30, 33, 34

DAA Digital Advertising Alliance. 75

DHCP Dynamic Host Configuration Protocol. 78

DN Distinguished Name. 72

DNS Domain Name System. 66, 67, 70, 77, 104

DoH DNS over HTTPS. 104

DV Domain Validated. 72, 73

GDPR General Data Protection Regulation. 18, 19, 29, 72, 75, 108

HbbTV Hybrid broadcast broadband TV. 98–100

HCI Human-Computer Interaction. 5

HTTP Hypertext Transfer Protocol. 63, 66, 67, 78, 87, 90, 98

IAB Interactive Advertising Bureau. 75, 91

IANA Internet Assigned Numbers Authority. 71

ICANN Internet Corporation for Assigned Names and Numbers. 72

IEEE Institute of Electrical and Electronics Engineers. 78

InfoVis Information Visualisation. 5

IoT Internet of Things. ix, xi, 1, 3, 7, 13, 14, 27, 34, 44, 104, 109, 143, 144

IP Internet Protocol. 63, 65, 67–70, 77, 78

IPA Intelligent Personal Assistant. 18, 30

ISP Internet Service Provider. 70

JKU Johannes Kepler University. 95

JNA Java Native Access. 78

JSON JavaScript Object Notation. 70

MAC Media Access Control. 3, 77, 78

MITM Man-In-The-Middle. 63, 77

MTU Maximum Transmission Unit. 65

MUD Manufacturer Usage Description. 43, 109

NSM Network Security Monitor. 63

ODM Open Data Map. 73

OTT Over-The-Top. 108

OUI Organizationally Unique Identifier. 78

PD Participatory Design. ix, xi, 6, 7, 10–12

PET Privacy-Enhancing Technologies. 36, 109

PII Personally Identifiable Information. 63, 72

PSL Public Suffix List. 68

RR Resource Record. 67

SNI Server Name Indication. 67, 104

SPF Sender Policy Framework. 70

SSDP Simple Service Discovery Protocol. 78

SSH Secure Shell. 90

TCF Transparency & Consent Framework. 75

TCP Transmission Control Protocol. 63, 65–67

TET Transparency Enhancing Tool. 109

TLD Top-Level-Domain. 68, 71

TLS Transport Layer Security. 66, 67, 72, 73, 87, 104

UPnP Universal Plug and Play. 78, 79

URI Uniform Resource Identifier. 66, 67, 90

VOD Video on Demand. 98–100

Bibliography

[1] AAX LLC. *AAX - Acceptable Ads Exchange*. 2019. URL: https://aax.media/ (accessed 08/31/2019).

[2] Noura Abdi, Kopo M Ramokapane, and Jose M Such. "More than Smart Speakers: Security and Privacy Perceptions of Smart Home Personal Assistants". In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. 2019.

[3] Steve Alexander and Ralph Droms. *DHCP Options and BOOTP Vendor Extensions*. RFC 2132. RFC Editor, Mar. 1997. URL: https://www.rfc-editor.org/rfc/rfc2132.txt.

[4] Amazon. *AWS IP Address Ranges*. 2019. URL: https://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html (accessed 11/23/2019).

[5] Noah Apthorpe, Yan Shvartzshnaider, Arunesh Mathur, Dillon Reisman, and Nick Feamster. "Discovering smart home internet of things privacy norms using contextual integrity". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.2 (2018), p. 59.

[6] Noah Apthorpe, Sarah Varghese, and Nick Feamster. "Evaluating the Contextual Integrity of Privacy Regulation: Parents' IoT Toy Privacy Norms Versus COPPA". In: *28th USENIX Security Symposium (USENIX Security 19)*. Aug. 2019, pp. 123–140.

[7] Maggie Astor. "Your Roomba May Be Mapping Your Home, Collecting Data That Could Be Shared". In: *The New York Times* (July 25, 2017). URL: https://www.nytimes.com/2017/07/25/technology/roomba-irobot-data-privacy.html (accessed 08/31/2019).

[8] Andrei Banaru. "Philips Hue 2.1 – Enabling WIFI". In: *IoT Blog* (Dec. 20, 2018). URL: https://blog.andreibanaru.ro/2018/03/27/philips-hue-2-1-enabling-wifi/ (accessed 08/31/2019).

[9] BBC News. "John Hancock adds fitness tracking to all policies". In: *BBC News* (Sept. 20, 2018). URL: https://www.bbc.com/news/technology-45590293 (accessed 08/31/2019).

[10] Richard Bejtlich. "Chapter 2 - Collecting Network Traffic Access, Storage and Management". In: *The practice of network security monitoring: understanding incident detection and response*. No Starch Press, 2013.

[11] Tim Berners-Lee, Roy T. Fielding, and Larry Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. STD 66. RFC Editor, Jan. 2005. URL: https://www.rfc-editor.org/rfc/rfc3986.txt.

[12] Sam Biddle. "For Owners of Amazon's Ring Security Cameras, Stranger May Have Been Watching Too". In: *The Intercept* (Jan. 10, 2019). URL: https://theintercept.com/2019/01/10/amazon-ring-security-camera/ (accessed 08/31/2019).

[13] Reuben Binns, Jun Zhao, Max Van Kleek, Nigel Shadbolt, Ilaria Liccardi, and Daniel Weitzner. "My bank already gets this data: Exposure minimisation and company relationships in privacy decision-making". In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM. 2017, pp. 2403–2409.

[14] Simon Blake-Wilson, Magnus Nystrom, David Hopwood, Jan Mikkelsen, and Tim Wright. *Transport Layer Security (TLS) Extensions*. RFC 3546. RFC Editor, June 2003. URL: https://www.rfc-editor.org/rfc/rfc3546.txt.

[15] Nellie Bowles. "Thermostats, Locks and Lights: Digital Tools of Domestic Abuse". In: *The New York Times* (June 23, 2018). URL: https://www.nytimes.com/2018/06/23/technology/smart-home-devices-domestic-abuse.html (accessed 08/31/2019).

[16] Sam Byford. "Apple stops letting contractors listen to Siri voice recordings and will offer opt-out later". In: *The Verge* (Aug. 2, 2019). URL: https://www.theverge.com/2019/8/2/20751270/apple-stops-contractors-siri-voice-recordings-privacy-opt-out (accessed 08/31/2019).

[17] CAIDA. *The CAIDA AS Organizations Dataset*. 2019. URL: http://www.caida.org/data/as-organizations/ (accessed 11/23/2019).

[18] Simone Carletti. *Whois, an intelligent — pure Ruby — WHOIS client and parser*. 2019. URL: https://whoisrb.org/ (accessed 11/23/2019).

[19] Marshini Chetty, Hyojoon Kim, Srikanth Sundaresan, Sam Burnett, Nick Feamster, and W Keith Edwards. "ucap: An internet data management tool for the home". In: *Proceedings of the 33rd annual ACM conference on human factors in computing systems*. ACM. 2015, pp. 3093–3102.

[20] John Cook, Rishab Nithyanand, and Zubair Shafiq. "Inferring Tracker-Advertiser Relationships in the Online Advertising Ecosystem using Header Bidding". In: *Proceedings on Privacy Enhancing Technologies* 1 (2020), pp. 65–82.

[21] David Cooper, Stefan Santesson, Stephen Farrell, Sharon Boeyen, Russell Housley, and Tim Polk. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. RFC 5280. RFC Editor, May 2008. URL: https://www.rfc-editor.org/rfc/rfc5280.txt.

[22] Vicka Corey, Charles Peterman, Sybil Shearin, Michael S Greenberg, and James Van Bokkelen. "Network forensics analysis". In: *Internet Computing, IEEE* 6.6 (2002), pp. 60–66.

[23] Council of European Union. *GDPR - Article 12 - Transparent information, communication and modalities for the exercise of the rights of the data subject*. 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj#d1e2172-1-1.

[24] Leslie Daigle. *WHOIS Protocol Specification*. RFC 3912. RFC Editor, Sept. 2004. URL: https://www.rfc-editor.org/rfc/rfc3912.txt.

[25] Sherri Davidoff and Jonathan Ham. In: *Network forensics: tracking hackers through cyberspace*. Prentice hall, 2012. Chap. Evidence Acquisition, pp. 45–72.

[26] Matt Day. "Amazon Gives Option to Disable Human Review on Alexa". In: *Bloomberg* (Aug. 3, 2019). URL: https://www.bloomberg.com/news/articles/2019-08-02/amazon-gives-option-to-disable-human-review-of-alexa-recordings (accessed 08/31/2019).

[27] Matt Day. "Amazon Is Working on a Device That Can Read Human Emotions". In: *Bloomberg* (May 23, 2019). URL: https://www.bloomberg.com/news/articles/2019-05-23/amazon-is-working-on-a-wearable-device-that-reads-human-emotions (accessed 08/31/2019).

[28] Matt Day, Giles Turner, and Natalia Drozdiak. "Amazon Workers Are Listening to What You Tell Alexa". In: *Bloomberg* (Apr. 11, 2019). URL: https://www.bloomberg.com/news/articles/2019-04-10/is-anyone-listening-to-you-on-alexa-a-global-team-reviews-audio (accessed 08/31/2019).

[29] Tim Dierks and Eric Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246. RFC Editor, Aug. 2008. URL: https://www.rfc-editor.org/rfc/rfc5246.txt.

[30] Digital Advertising Alliance. *WebChoices: Digital Advertising Alliance's Consumer Choice Tool*. 2019. URL: http://optout.aboutads.info/ (accessed 11/23/2019).

[31] Disconnect. *Tracker Protection Info*. 2019. URL: https://disconnect.me/trackerprotection (accessed 11/23/2019).

[32] DNSFilter. *Transparent Proxying*. 2019. URL: https://docs.dnsfilter.com/docs/transparent-proxying (accessed 11/23/2019).

[33] Markus Donko-Huber. *Upribox*. 2019. URL: https://upribox.org/ (accessed 08/31/2019).

[34] Ralph Droms. *Dynamic Host Configuration Protocol*. RFC 2131. RFC Editor, Mar. 1997. URL: https://www.rfc-editor.org/rfc/rfc2131.txt.

[35] DuckDuckGo. *DuckDuckGo Tracker Radar Exposes Hidden Tracking*. Mar. 5, 2020. URL: https://spreadprivacy.com/duckduckgo-tracker-radar/ (accessed 03/05/2020).

[36] Simon Elvery. “My phone is spying on me, so I decided to spy on it”. In: *ABC News* (Dec. 3, 2018). URL: https://www.abc.net.au/news/2018-10-25/my-phone-is-spying-on-me-so-i-decided-to-spy-on-my-phone/10306586 (accessed 08/31/2019).

[37] Pardis Emami-Naeini, Yuvraj Agarwal, Lorrie Faith Cranor, and Hanan Hibshi. *Ask the Experts: What Should Be on an IoT Privacy and Security Label?* 2020. arXiv: 2002.04631 [cs.CY].

[38] Pardis Emami-Naeini, Sruti Bhagavatula, Hana Habib, Martin Degeling, Lujo Bauer, Lorrie Faith Cranor, and Norman Sadeh. “Privacy expectations and preferences in an IoT world”. In: *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. 2017, pp. 399–412.

[39] Pardis Emami-Naeini, Henry Dixon, Yuvraj Agarwal, and Lorrie Faith Cranor. “Exploring how privacy and security factor into IoT device purchase behavior”. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM. 2019, p. 534.

[40] Evidon. *Global Consent Preferences*. 2019. URL: https://www.evidon.com/resources/global-opt-out/ (accessed 11/23/2019).

[41] Exodus Privacy. *ϵxodus - The privacy audit platform for Android applications*. 2019. URL: https://reports.exodus-privacy.eu.org/en/ (accessed 11/23/2019).

[42] Lesley Fair. “What Vizio was doing behind the TV screen”. In: *Federal Trade Commission* (Feb. 6, 2017). URL: https://www.ftc.gov/news-events/blogs/business-blog/2017/02/what-vizio-was-doing-behind-tv-screen (accessed 08/31/2019).

[43] Fastly. *TLS service options - Shared TLS Certificate Service*. 2019. URL: https://docs.fastly.com/products/tls-service-options#shared-tls-certificate-service (accessed 11/23/2019).

[44] Roy T. Fielding, James Gettys, Jeffrey C. Mogul, Henrik Frystyk Nielsen, Larry Masinter, Paul J. Leach, and Tim Berners-Lee. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616. RFC Editor, June 1999. URL: https://www.rfc-editor.org/rfc/rfc2616.txt.

[45] Forbrukerrådet. “Out of control: How consumers are exploited by the online advertising industry”. In: (Jan. 14, 2020). URL: https://www.forbrukerradet.no/undersokelse/no-undersokelsekategori/report-out-of-control/.

[46] Geoffrey A. Fowler. “The spy in your wallet: Credit cards have a privacy problem”. In: *The Washington Post* (Aug. 26, 2019). URL: https://www.washingtonpost.com/technology/2019/08/26/spy-your-wallet-credit-cards-have-privacy-problem/ (accessed 08/31/2019).

[47] Geoffrey A. Fowler. “You watch TV. Your TV watches back.” In: *The Washington Post* (Sept. 18, 2019). URL: https://www.washingtonpost.com/technology/2019/09/18/you-watch-tv-your-tv-watches-back/ (accessed 12/08/2019).

[48] Fraunhofer FOKUS. *HbbTV-based Broadcast Measurement and TV Research Tool*. 2017. URL: https://www.fokus.fraunhofer.de/en/fame/hbbtv-bm (accessed 02/01/2020).

[49] Sarah Frier. “Facebook Paid Contractors to Transcribe Users’ Audio Chats”. In: *Bloomberg* (Aug. 13, 2019). URL: https://www.bloomberg.com/news/articles/2019-08-13/facebook-paid-hundreds-of-contractors-to-transcribe-users-audio (accessed 08/31/2019).

[50] Jon Froehlich, Leah Findlater, Marilyn Ostergren, Solai Ramanathan, Josh Peterson, Inness Wragg, Eric Larson, Fabia Fu, Mazhengmin Bai, Shwetak Patel, and James Landay. “The design and evaluation of prototype eco-feedback displays for fixture-level water usage data”. In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2012, pp. 2367–2376.

[51] Sidney Fussell. “Airbnb Has a Hidden-Camera Problem”. In: *The Atlantic* (Mar. 26, 2019). URL: https://www.theatlantic.com/technology/archive/2019/03/what-happens-when-you-find-cameras-your-airbnb/585007/ (accessed 02/01/2020).

[52] Gartner. “Internet of Things endpoint spending worldwide by category from 2014 to 2020 (in billion U.S. dollars) [Graph]”. In: *Statista* (Feb. 7, 2017). URL: https://www.statista.com/statistics/485252/iot-endpoint-spending-by-category-worldwide/ (accessed 08/31/2019).

[53] Marco Ghiglieri, Melanie Volkamer, and Karen Renaud. “Exploring consumers’ attitudes of smart TV related privacy risks”. In: *International Conference on Human Aspects of Information Security, Privacy, and Trust*. Springer. 2017, pp. 656–674.

[54] Marco Ghiglieri and Michael Waidner. “HbbTV security and privacy: issues and challenges”. In: *IEEE Security & Privacy* 14.3 (2016), pp. 61–67.

[55] GlassWire. *GlassWire - Personal Firewall & Network Monitor*. 2019. URL: https://www.glasswire.com (accessed 08/31/2019).

[56] GoDaddy. *DV SSL certificates: a trusted level of domain validation*. 2019. URL: https://www.godaddy.com/web-security/domain-validation-ssl-certificate (accessed 11/23/2019).

[57] Google. *Google Compute Engine FAQ - Where can I find Compute Engine IP ranges?* 2019. URL: https://cloud.google.com/compute/docs/faq#find_ip_range (accessed 11/23/2019).

[58] Ayyoob Hamza, Dinesha Ranathunga, Hassan Habibi Gharakheili, Theophilus A Benson, Matthew Roughan, and Vijay Sivaraman. "Verifying and monitoring iots network behavior using mud profiles". In: *arXiv preprint arXiv:1902.02484* (2019).

[59] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. "Polisis: Automated analysis and presentation of privacy policies using deep learning". In: *27th USENIX Security Symposium (USENIX Security 18)*. 2018, pp. 531–548.

[60] John Hawkinson and Tony Bates. *Guidelines for creation, selection, and registration of an Autonomous System (AS)*. BCP 6. RFC Editor, Mar. 1996. URL: https://www.rfc-editor.org/rfc/rfc1930.txt.

[61] Alex Hern. "Apple contractors 'regularly hear confidential details' on Siri recordings". In: *The Guardian* (July 26, 2019). URL: https://www.theguardian.com/technology/2019/jul/26/apple-contractors-regularly-hear-confidential-details-on-siri-recordings (accessed 08/31/2019).

[62] Paul Hoffman and Patrick McManus. *DNS Queries over HTTPS (DoH)*. RFC 8484. RFC Editor, Aug. 2018. URL: https://www.rfc-editor.org/rfc/rfc8484.txt.

[63] Pi-hole. *Announcing a Beta test of Pi-hole 5.0!* 2020. URL: https://pi-hole.net/2020/01/19/announcing-a-beta-test-of-pi-hole-5-0 (accessed 02/01/2020).

[64] Pi-hole. *Pi-hole: A black hole for Internet advertisments*. 2019. URL: https://pi-hole.net/ (accessed 08/31/2019).

[65] Pi-hole. *Pi-hole Documentation - Blocking mode*. 2019. URL: https://docs.pi-hole.net/ftldns/blockingmode/ (accessed 11/23/2019).

[66] Jason Hong. "The privacy landscape of pervasive computing". In: *IEEE Pervasive Computing* 16.3 (2017), pp. 40–48.

[67] Danny Yuxing Huang, Noah Apthorpe, Gunes Acar, Frank Li, and Nick Feamster. *IoT Inspector: Crowdsourcing Labeled Network Traffic from Smart Home Devices at Scale*. 2019. arXiv: 1909.09848 [cs.CR].

[68] IAB. *TCF – Transparency & Consent Framework*. 2019. URL: https://iabeurope.eu/tcf-2-0/ (accessed 11/23/2019).

[69] IANA. *Autonomous System (AS) Numbers*. 2019. URL: https://www.iana.org/assignments/as-numbers/ (accessed 11/23/2019).

[70] IANA. *Root Zone Database*. 2019. URL: https://www.iana.org/domains/root/db (accessed 11/23/2019).

[71] IANA. *Service Name and Transport Protocol Port Number Registry*. 2019. URL: https://www.iana.org/assignments/service-names-port-numbers/ (accessed 08/31/2019).

[72] ICANN. *Temporary Specification for gTLD Registration Data*. 2019. URL: https://www.icann.org/resources/pages/gtld-registration-data-specs-en (accessed 11/23/2019).

[73] ICANN. *What are thick and thin entries?* 2019. URL: https://whois.icann.org/en/what-are-thick-and-thin-entries (accessed 11/23/2019).

[74] IEEE. *OUI*. 2019. URL: http://standards-oui.ieee.org/oui.txt (accessed 11/23/2019).

[75] Privacy International. "How Apps on Android Share Data with Facebook (even if you don't have a Facebook account)". In: *Privacy International* (Dec. 2018). URL: https://privacyinternational.org/report/2647/how-apps-android-share-data-facebook-report.

[76] Tobias Isenberg, Petra Isenberg, Jian Chen, Michael Sedlmair, and Torsten Möller. "A systematic review on the practice of evaluating visualization". In: *IEEE Transactions on Visualization and Computer Graphics* 19.12 (2013), pp. 2818–2827.

[77] ISO. *ISO 3166 Country Codes*. 2019. URL: https://www.iso.org/iso-3166-country-codes.html (accessed 11/23/2019).

[78] JHipster. *JHipster - Generate your Spring Boot + Angular/React application*. 2020. URL: https://www.jhipster.tech/ (accessed 03/01/2020).

[79] Haojian Jin, Minyi Liu, Kevan Dodhia, Yuanchun Li, Gaurav Srivastava, Matthew Fredrikson, Yuvraj Agarwal, and Jason I Hong. "Why Are They Collecting My Data?: Inferring the Purposes of Network Traffic in Mobile Apps". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.4 (2018), p. 173.

[80] Arjaldo Karaj, Sam Macbeth, Rémi Berson, and Josep M. Pujol. *WhoTracks.Me: Shedding light on the opaque world of online tracking*. 2018. arXiv: 1804.08959 [cs.CY].

[81] Farzaneh Karegar, Tobias Pulls, and Simone Fischer-Hübner. "Visualizing Exports of Personal Data by Exercising the Right of Data Portability in the Data Track-Are People Ready for This?" In: *IFIP International Summer School on Privacy and Identity Management*. Springer. 2016, pp. 164–181.

[82] Simon Kelley. *Dnsmasq - A lightweight DHCP and caching DNS server*. 2018. URL: http://www.thekelleys.org.uk/dnsmasq/docs/dnsmasq-man.html (accessed 11/23/2019).

[83] Steve Kille. *A String Representation of Distinguished Names*. RFC 1779. RFC Editor, Mar. 1995. URL: https://www.rfc-editor.org/rfc/rfc1779.txt.

[84] Thorin Klosowski. "Why You Should Take a Close Look at What Tracks You". In: *The New York Times* (Jan. 7, 2020). URL: https://www.nytimes.com/2020/01/07/opinion/location-tracking-privacy.html (accessed 02/01/2020).

[85] Charles M Kozierok. *The TCP/IP guide: a comprehensive, illustrated Internet protocols reference*. No Starch Press, 2005.

[86] Vinayshekhar Bannihatti Kumar, Roger Iyengar, Namita Nisal, Yuanyuan Feng, Hana Habib, Peter Story, Sushain Cherivirala, Margaret Hagan, Lorrie Faith Cranor, Shomir Wilson, et al. "Finding a Choice in a Haystack: Automatic Extraction of Opt-Out Statements from Privacy Policy Text". In: *The Web Conference (the Web Conf)*. 2020.

[87] Josephine Lau, Benjamin Zimmerman, and Florian Schaub. "Alexa, are you listening?: Privacy perceptions, concerns and privacy-seeking behaviors with smart speakers". In: *Proceedings of the ACM on Human-Computer Interaction* 2.CSCW (2018), p. 102.

[88] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

[89] Eliot Lear, Ralph Droms, and Dan Romascanu. *Manufacturer Usage Description Specification*. RFC 8520. RFC Editor, Mar. 2019. URL: https://www.rfc-editor.org/rfc/rfc8520.txt.

[90] Timothy Libert. "An Automated Approach to Auditing Disclosure of Third-Party Data Collection in Website Privacy Policies". In: *Proceedings of the 2018 World Wide Web Conference*. WWW '18. ACM. 2018, pp. 207–216.

[91] Lifewire. *What It Means When You See the 0.0.0.0 IP Address*. 2019. URL: https://www.lifewire.com/four-zero-ip-address-818384 (accessed 11/23/2019).

[92] Jialiu Lin, Shahriyar Amini, Jason I Hong, Norman Sadeh, Janne Lindqvist, and Joy Zhang. "Expectation and purpose: understanding users' mental models of mobile app privacy through crowdsourcing". In: *Proceedings of the 2012 ACM conference on ubiquitous computing*. ACM. 2012, pp. 501–510.

[93] LUMA Partners. *Display LUMAscape*. 2019. URL: https://lumapartners.com/content/lumascapes/display-ad-tech-lumascape/ (accessed 08/31/2019).

[94] Sapna Maheshwari. "Hey, Alexa, What Can You Hear? And What Will You Do With It?" In: *The New York Times* (Mar. 31, 2018). URL: https://www.nytimes.com/2018/03/31/business/media/amazon-google-privacy-digital-assistants.html (accessed 08/31/2019).

[95] Sapna Maheshwari. "How Smart TVs in Millions of U.S. Homes Track More Than What's On Tonight". In: *The New York Times* (July 5, 2018). URL: https://www.nytimes.com/2018/07/05/business/media/tv-viewer-tracking.html (accessed 08/31/2019).

[96] Sapna Maheshwari. “That Game on Your Phone May Be Tracking What You’re Watching on TV”. In: *The New York Times* (Dec. 28, 2017). URL: https://www.nytimes.com/2017/12/28/business/media/alphonso-app-tracking.html (accessed 08/31/2019).

[97] Nathan Malkin, Julia Bernd, Maritza Johnson, and Serge Egelman. “What Can’t Data Be Used For?” In: *Privacy Expectations about Smart TVs in the US. European Workshop on Usable Security (EuroUSEC)*. 2018.

[98] Nathan Malkin, Joe Deatrick, Allen Tong, Primal Wijesekera, Serge Egelman, and David Wagner. “Privacy Attitudes of Smart Speaker Users”. In: *Proceedings on Privacy Enhancing Technologies* 2019.4 (2019), pp. 250–271.

[99] MaxMind. *GeoLite2 Databases*. 2019. URL: https://dev.maxmind.com/geoip/geoip2/geolite2/ (accessed 11/23/2019).

[100] Sarah Mennicken, David Kim, and Elaine May Huang. “Integrating the smart home into the digital calendar”. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM. 2016, pp. 5958–5969.

[101] Georg Merzdovnik, Markus Huber, Damjan Buhov, Nick Nikiforakis, Sebastian Neuner, Martin Schmiedecker, and Edgar Weippl. “Block me if you can: A large-scale study of tracker-blocking tools”. In: *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. 2017, pp. 319–333.

[102] Microsoft. *Microsoft Azure Datacenter IP Ranges*. 2019. URL: https://www.microsoft.com/en-us/download/details.aspx?id=41653 (accessed 11/23/2019).

[103] Paul Mockapetris. *Domain names - concepts and facilities*. STD 13. RFC Editor, Nov. 1987. URL: https://www.rfc-editor.org/rfc/rfc1034.txt.

[104] Paul Mockapetris. *Domain names - implementation and specification*. STD 13. RFC Editor, Nov. 1987. URL: https://www.rfc-editor.org/rfc/rfc1035.txt.

[105] Jeffrey Mogul and Steve Deering. *Path MTU discovery*. RFC 1191. RFC Editor, Nov. 1990. URL: https://www.rfc-editor.org/rfc/rfc1191.txt.

[106] Hooman Mohajeri Moghaddam, Gunes Acar, Ben Burgess, Arunesh Mathur, Danny Yuxing Huang, Nick Feamster, Edward W Felten, Prateek Mittal, and Arvind Narayanan. “Watching You Watch: The Tracking Ecosystem of Over-the-Top TV Streaming Devices”. In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2019, pp. 131–147.

[107] Mozilla. *Firefox continues push to bring DNS over HTTPS by default for US users*. Feb. 25, 2020. URL: https://blog.mozilla.org/blog/2020/02/25/firefox-continues-push-to-bring-dns-over-https-by-default-for-us-users/ (accessed 02/25/2020).

[108] Mozilla. *privacy not included*. 2019. URL: https://foundation.mozilla.org/en/privacynotincluded/ (accessed 08/31/2019).

[109] Mozilla. *Public Suffix List*. 2019. URL: https://publicsuffix.org/ (accessed 11/23/2019).

[110] Tamara Munzner. "A nested model for visualization design and validation". In: *IEEE transactions on visualization and computer graphics* 15.6 (2009), pp. 921–928.

[111] Patrick Murmann and Simone Fischer-Hübner. "Tools for achieving usable ex post transparency: a survey". In: *IEEE Access* 5 (2017), pp. 22965–22991.

[112] Helen Nissenbaum. "Privacy as contextual integrity". In: *Wash. L. Rev.* 79 (2004), p. 119.

[113] NOYB – European Center for Digital Rights. "Three GDPR Complaints filed against Grindr, Twitter and the AdTech companies Smaato, OpenX, AdColony and AT&T's AppNexus". In: (Jan. 14, 2020). URL: https://noyb.eu/en/three-gdpr-complaints-filed-against-grindr-twitter-and-adtech-companies-smaato-openx-adcolony-and (accessed 05/08/2020).

[114] Objective Development Software GmbH. *Little Snitch 4*. 2017. URL: https://www.obdev.at/products/littlesnitch/index.html (accessed 08/31/2019).

[115] Philip Oltermann. "German parents told to destroy doll that can spy on children". In: *The Guardian* (Feb. 17, 2017). URL: https://www.theguardian.com/world/2017/feb/17/german-parents-told-to-destroy-my-friend-cayla-doll-spy-on-children (accessed 08/31/2019).

[116] Antti Oulasvirta and Kasper Hornbæk. "Hci research as problem-solving". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM. 2016, pp. 4956–4967.

[117] Xinru Page, Paritosh Bahirat, Muhammad I Safi, Bart P Knijnenburg, and Pamela Wisniewski. "The Internet of What?: Understanding Differences in Perceptions and Adoption for the Internet of Things". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.4 (2018), p. 183.

[118] Elleen Pan, Jingjing Ren, Martina Lindorfer, Christo Wilson, and David Choffnes. "Panoptispy: Characterizing audio and video exfiltration from android applications". In: *Proceedings on Privacy Enhancing Technologies* 2018.4 (2018), pp. 33–50.

[119] Nilay Patel. "Taking the smarts out of smart TVs would make them more expensive". In: *The Verge* (Jan. 7, 2019). URL: https://www.theverge.com/2019/1/7/18172397/airplay-2-homekit-vizio-tv-bill-baxter-interview-vergecast-ces-2019 (accessed 08/31/2019).

[120] Vern Paxson. "Bro: a system for detecting network intruders in real-time". In: *Computer networks* 31.23-24 (1999), pp. 2435–2463.

[121] Tim Peterson. “Roku’s advertising business is outpacing its hardware business”. In: *Digiday* (Aug. 8, 2018). URL: https://digiday.com/media/rokus-advertising-business-outpacing-hardware-business/ (accessed 08/31/2019).

[122] Antigone Peyton. “A Litigator’s Guide to the Internet of Things”. In: *Richmond Journal of Law & Technology* 22.3 (2016), p. 9.

[123] Jon Postel. *Internet Protocol*. STD 5. RFC Editor, Sept. 1981. URL: https://www.rfc-editor.org/rfc/rfc791.txt.

[124] Jon Postel. *Transmission Control Protocol*. STD 7. RFC Editor, Sept. 1981. URL: https://www.rfc-editor.org/rfc/rfc793.txt.

[125] Princeton University. *Princeton IoT Inspector*. 2019. URL: https://iot-inspector.princeton.edu/ (accessed 08/31/2019).

[126] Ramona Pringle. “’It’s time for us to watch them’: App lets you spy on Alexa and the rest of your smart devices”. In: *CBC News* (Apr. 25, 2019). URL: https://www.cbc.ca/news/technology/pringle-smart-home-privacy-1.5109347 (accessed 08/31/2019).

[127] Privacy International. “How do data companies get our data?” In: *Privacy International* (May 25, 2018). URL: https://medium.com/@privacyint/how-do-data-companies-get-our-data-fc2aec6101d6 (accessed 05/08/2020).

[128] Abbas Razaghpanah, Rishab Nithyanand, Narseo Vallina-Rodriguez, Srikanth Sundaresan, Mark Allman, and Christian Kreibich Phillipa Gill. “Apps, trackers, privacy, and regulators”. In: *25th Annual Network and Distributed System Security Symposium, NDSS*. Vol. 2018. 2018.

[129] Abbas Razaghpanah, Narseo Vallina-Rodriguez, Srikanth Sundaresan, Christian Kreibich, Phillipa Gill, Mark Allman, and Vern Paxson. *Haystack: A Multi-Purpose Mobile Vantage Point in User Space*. 2015. arXiv: 1510.01419 [cs.NI].

[130] Jingjing Ren, Daniel J Dubois, David Choffnes, Anna Maria Mandalari, Roman Kolcun, and Hamed Haddadi. “Information Exposure From Consumer IoT Devices: A Multidimensional, Network-Informed Measurement Approach”. In: *Proceedings of the Internet Measurement Conference*. ACM. 2019, pp. 267–279.

[131] Jingjing Ren, Ashwin Rao, Martina Lindorfer, Arnaud Legout, and David Choffnes. “Recon: Revealing and controlling pii leaks in mobile network traffic”. In: *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. ACM. 2016, pp. 361–374.

[132] Eric Rescorla. *HTTP Over TLS*. RFC 2818. RFC Editor, May 2000. URL: https://www.rfc-editor.org/rfc/rfc2818.txt.

[133] Eric Rescorla, Kazuho Oku, Nick Sullivan, and Christopher Wood. *Encrypted Server Name Indication for TLS 1.3*. Internet-Draft draft-ietf-tls-esni-05. IETF Secretariat, Nov. 2019. URL: https://datatracker.ietf.org/doc/draft-ietf-tls-esni/.

[134] Gilad Rosner. *Privacy and the Internet of Things*. O'Reilly Media, 2016.

[135] Gilad Rosner and Erin Kenneally. "Clearly Opaque: Privacy Risks of the Internet of Things". In: *Rosner, Gilad and Kenneally, Erin, Clearly Opaque: Privacy Risks of the Internet of Things (May 1, 2018). IoT Privacy Forum*. 2018.

[136] Jeffrey Rubin and Dana Chisnell. *Handbook of usability testing: how to plan, design and conduct effective tests*. John Wiley & Sons, 2008.

[137] Miriam Ruhenstroth. "How Facebook knows which apps you use – and why this matters". In: *mobilsicher.de* (Dec. 20, 2018). URL: https://mobilsicher.de/hintergrund/how-facebook-knows-which-apps-you-use-and-why-this-matters (accessed 08/31/2019).

[138] Johnny Saldaña. *The coding manual for qualitative researchers*. Sage, 2015.

[139] Chris Sanders and Jason Smith. "Chapter 3 - The Sensor Platform". In: *Applied network security monitoring: collection, detection, and analysis*. Elsevier, 2013.

[140] Florian Schaub, Aditya Marella, Pranshu Kalvani, Blase Ur, Chao Pan, Emily Forney, and Lorrie Faith Cranor. "Watching them watching me: Browser extensions' impact on user privacy awareness and concern". In: *NDSS workshop on usable security*. 2016.

[141] Martin Schemm. "Speech assistance systems put to the test - Data protection authority opens administrative proceedings against Google". In: *Datenschutz Hamburg* (Aug. 1, 2019). URL: https://datenschutz-hamburg.de/assets/pdf/2019-08-01_press-release-Google_Assistant.pdf (accessed 08/31/2019).

[142] Michael Sedlmair, Miriah Meyer, and Tamara Munzner. "Design study methodology: Reflections from the trenches and the stacks". In: *IEEE transactions on visualization and computer graphics* 18.12 (2012), pp. 2431–2440.

[143] ServusTV. *ServusTV Imprint*. 2020. URL: https://richtlinien.servus.com/policies/Servus/202002261130/de/imprint.html (accessed 02/01/2020).

[144] William Seymour, Martin J. Kraemer, Reuben Binns, and Max Van Kleek. *Informing the Design of Privacy-Empowering Tools for the Connected Home*. 2020. arXiv: 2001.09077 [cs.HC].

[145] Ben Shneiderman. "The eyes have it: A task by data type taxonomy for information visualizations". In: *Proceedings 1996 IEEE symposium on visual languages*. IEEE. 1996, pp. 336–343.

[146] Manuel Silverio-Fernández, Suresh Renukappa, and Subashini Suresh. "What is a smart device? - a conceptualisation within the paradigm of the internet of things". In: *Visualization in Engineering* 6.1 (2018), p. 3.

[147] Small Technology Foundation. *Better Blocker - a privacy tool for Safari on iPhone, iPad, and Mac*. 2019. URL: https://better.fyi/ (accessed 11/23/2019).

[148] Robin Sommer. “Bro: An Open Source Network Intrusion Detection System.” In: *DFN-Arbeitstagung über Kommunikationsnetze*. Citeseer. 2003, pp. 273–288.

[149] Gaurav Srivastava, Kunal Bhuwalka, Swarup Kumar Sahoo, Saksham Chitkara, Kevin Ku, Matt Fredrikson, Jason Hong, and Yuvraj Agarwal. *PrivacyProxy: Leveraging Crowdsourcing and In Situ Traffic Analysis to Detect and Mitigate Information Leakage*. 2017. arXiv: 1708.06384 [cs.CR].

[150] Madiha Tabassum, Tomasz Kosinski, and Heather Richter Lipford. “"I don’t own the data": End User Perceptions of Smart Home Device Data Practices and Risks”. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. 2019.

[151] The Digital Standard. *The Digital Standard*. 2019. URL: https://www.thedigitalstandard.org (accessed 08/31/2019).

[152] The Tcpdump Group. *Tcpdump & Libpcap*. 2020. URL: https://www.tcpdump.org/ (accessed 03/01/2020).

[153] Yuan Tian, Nan Zhang, Yueh-Hsun Lin, XiaoFeng Wang, Blase Ur, Xianzheng Guo, and Patrick Tague. “Smartauth: User-centered authorization for the internet of things”. In: *26th USENIX Security Symposium (USENIX Security 17)*. 2017, pp. 361–378.

[154] TrustArc. *TrustArc Preference Manager*. 2019. URL: https://preferences-mgr.truste.com/ (accessed 11/23/2019).

[155] Matteo Turilli and Luciano Floridi. “The ethics of information transparency”. In: *Ethics and Information Technology* 11.2 (2009), pp. 105–112.

[156] UPnP Forum. *UPnP Device Architecture 1.1*. 2008. URL: http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf (accessed 11/23/2019).

[157] Lente Van Hee, Ruben Van Den Heuvel, Tim Verheyden, and Denny Baert. “Google employees are eavesdropping, even in your living room, VRT NWS has discovered”. In: *VRT News* (July 10, 2019). URL: https://www.vrt.be/vrtnws/en/2019/07/10/google-employees-are-eavesdropping-even-in-flemish-living-rooms/ (accessed 08/31/2019).

[158] Max Van Kleek, Reuben Binns, Jun Zhao, Adam Slack, Sauyon Lee, Dean Ottewell, and Nigel Shadbolt. “X-ray refine: Supporting the exploration and refinement of information exposure resulting from smartphone apps”. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM. 2018, p. 393.

[159] Max Van Kleek, Ilaria Liccardi, Reuben Binns, Jun Zhao, Daniel J Weitzner, and Nigel Shadbolt. “Better the devil you know: Exposing the data sharing practices of smartphone apps”. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM. 2017, pp. 5208–5220.

[160] Janus Varmarken, Hieu Le, Anastasia Shuba, Athina Markopoulou, and Zubair Shafiq. “The TV is Smart and Full of Trackers: Measuring Smart TV Advertising and Tracking”. In: *Proceedings on Privacy Enhancing Technologies* 2 (2020), pp. 129–154.

[161] Susanne Weber, Marian Harbach, and Matthew Smith. “Participatory design for security-related user interfaces”. In: *Proc. USEC* 15 (2015).

[162] Ben Weinshel, Miranda Wei, Mainack Mondal, Euirim Choi, Shawn Shan, Claire Dolin, Michelle L Mazurek, and Blase Ur. “Oh, the Places You’ve Been! User Reactions to Longitudinal Transparency About Third-Party Web Tracking and Inferencing”. In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2019, pp. 149–166.

[163] Daricia Wilkinson, Paritosh Bahirat, Moses Namara, Jing Lyu, Arwa Alsubhi, Jessica Qiu, Pamela Wisniewski, and Bart P Knijnenburg. “Privacy at a Glance: The User-Centric Design of Glanceable Data Exposure Visualizations”. In: *Proceedings on Privacy Enhancing Technologies* 2 (2020), pp. 416–435.

[164] Wireshark. *OUI Lookup Tool*. 2019. URL: `https://www.wireshark.org/tools/oui-lookup.html` (accessed 11/23/2019).

[165] Wireshark. *Wireshark - Go Deep*. 2019. URL: `https://www.wireshark.org/` (accessed 11/23/2019).

[166] Dennis Wixon. “Evaluating usability methods: why the current literature fails the practitioner”. In: *interactions* 10.4 (2003), pp. 28–34.

[167] Meng Weng Wong and Wayne Schlitt. *Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1*. RFC 4408. RFC Editor, Apr. 2006. URL: `https://www.rfc-editor.org/rfc/rfc4408.txt`.

[168] XDA Developers. *Google Chromecast - Root Mini-FAQ*. 2016. URL: `https://forum.xda-developers.com/showthread.php?t=2621784` (accessed 08/31/2019).

[169] Yaxing Yao, Justin Reed Basdeo, Smirity Kaushik, and Yang Wang. “Defending My Castle: A Co-Design Study of Privacy Mechanisms for Smart Homes”. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM. 2019, p. 198.

[170] Eric Zeng, Shrirang Mare, and Franziska Roesner. “End user security and privacy concerns with smart homes”. In: *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. 2017, pp. 65–80.

[171] Eric Zeng and Franziska Roesner. “Understanding and Improving Security and Privacy in Multi-User Smart Homes: A Design Exploration and In-Home User Study”. In: *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 159–176. ISBN: 978-1-939133-06-9.

[172] Serena Zheng, Noah Apthorpe, Marshini Chetty, and Nick Feamster. “User perceptions of smart home IoT privacy”. In: *Proceedings of the ACM on Human-Computer Interaction* 2.CSCW (2018), p. 200.

Interview Script

This appendix contains the translated interview script used for the semi-structured interviews with 12 participants. Note that the script contains various background information, examples, and notes that were only gradually mentioned if participants had no ideas on their own, or if they desired some further context. This means that they were usually not mentioned at all, or only when participants had already given an answer themselves. Generally, great care was taken not to influence their responses in a predefined way, with more details about the interview procedure already discussed in Section 2.1.2.

Introduction and Demographic Information

- **Small talk** with technology tour through the apartment / the house, if performed at home
- Demographic **info**
  - Age, sex, profession / education
- **Background**: *A growing trend in households are so-called smart or IoT devices that connect to the Internet and often enable practical functionality such as automation, remote control, etc. However, it is becoming more and more complex for users to understand their communication behaviour (especially the outgoing one), since devices often communicate unknowingly in the background. This means that resulting dangers often remain invisible.*
- The focus / **intended goal** of my work is: *How to provide user-friendly insight into the communication behaviour of smart home IoT devices so that users feel informed about their behaviour.*
- Goals of the **interviews**:
  - Do you already use tools to gain insight into such communication behaviour?
  - How big is your awareness about dangers and unwanted background activities? No concern is also a valid answer.

- Do you have any concerns about purchasing or using certain smart devices?
- What information about the occurring data transfer would help you feel more informed?
- How could a usable visualisation of this information look like for you?
- And finally, how would you visualise your first ideas with a sketch?

- **Queries** are allowed if questions are not clear enough, or the reason for the question is not obvious.
- *Consent to the **recording** assuring their anonymity in the thesis*
- There are **no right or wrong answers**, or answers that I expect. The focus is on a requirements analysis.

Existing Tools

- (After the introduction:) Number and type of Internet-enabled **devices** in the household?
  - Usage scenarios
- Do you already use **tools** (software or hardware) to gain **insight** into the communication behaviour of devices, apps, or applications? *That means generally and not only focused on IoT devices.*
  - Tools that show the transfer to the Internet are of particular interest to me.
  - (Notes:) *Showing outgoing connections, personal firewall, IDS, etc*
  - (If yes:) Give a short description, live demo if possible
  - (If yes:) What were the reasons for using them?
    - (Examples:) *Recommendation, own interest, specific incident*
  - (If yes:) Which information did you expect to find about your devices?
  - (If yes:) Could you already gain new insight by using such a tool?
    - (Or:) Have you already been able to find out something worth knowing about your devices?
    - (If yes:) Has this changed your trust in the devices?
    - (If yes:) How is this reflected in your own behaviour?
    - (Example:) *Certain apps / features are no longer used because unwanted behaviour has been found.*
  - (If yes:) Are you missing certain information or insight that you would like to have?

        - How could the tool support you further?
    - (If yes:) Do you have general requests for improvements or problems when using the tool?
        - (Or:) Functional or usability problems? Limitations?
    - (If **no**:) Can you remember situations in which you would have liked to have such insight into the communication behaviour of your devices / applications?

- (If blocking connections has not yet been mentioned:) In addition to visibility, would you also like to actively **intervene** in the communication?
    - (If yes:) What would be reasons for you to intervene and to what extent?
    - (Example:) *Blocking certain connections*
    - (If yes:) Do you already use tools to intervene in the communication behaviour?
    - (Examples:) *adblocker, content filter, other measures*

- (If tools, but not for devices:) If you now think of tools specifically for **smart devices**: Would you have **special requirements** for them, compared to the tools already used?
    - Which features would be of high priority?
    - (Or:) What should a tool minimally show about devices to be of use?

- (Optional:) Do you tend to check or change the **settings** of your devices / applications before their first use? Apart from necessary configurations like wireless password.
    - (If yes:) Which and for what reason?
    - (Examples:) *Disable diagnostics data, change default password, disable unneeded features*

Awareness and Trust

- Have you been following news about threats from smart devices? (If yes:) Which example can you remember?
    - (If yes:) Did you find them exaggerated?
    - (If yes:) Where there any concerns that your own device might also be at risk?
    - (If yes:) Have there been changes in behaviour when using your devices?

- (Background:) *Internet-enabled toy ("My Friend Cayla" doll; British toy maker) was subsequently classified as an espionage tool in German by the Bundesnetzagentur and had to be provably destroyed. This means that even their possession is illegal, as it's officially a "hidden, transmitable radio."*

- If you think about the data transfer from devices to the Internet: Do you have any concerns about **purchasing** certain types of devices? (Similar:) Any concerns about **using** existing devices?
  - (Or:) What types of devices are you not using because of privacy concerns? Why?
  - (Example:) *Surveillance cameras that can film private areas in the home and can be hacked.*

- Are you concerned that smart devices may have **unknown background functionality**? That is, features that are not explicitly mentioned in the manual or in the settings.
  - (If yes:) Which ones?
  - (If yes:) For which devices or what kind of devices in particular? Why?
  - (If **no**:) Do you trust the manufacturer that the device will behave as expected?

- Which device behaviour would you consider **unwanted**? Can you give me examples? *Especially with regard to communication behaviour to the Internet.*

- (Optional:) When you think about unknown data collection and transfer: What **types of data** should never be transmitted from your devices to the Internet?
  - When you think about the **purpose** of the data collection: Does it affect your acceptance? Both positively and negatively.

- (If not yet asked:) Do you have an **implicit trust / distrust** in certain manufacturers with regard to device functionality?
  - (Similar:) Trust / distrust in **organisations** with regard to data processing?

If appropriate, mention a number of background information about unknown additional functionality.

- (Background:) *The viewing habits of certain smart TVs (Vizio) was transmitted to the manufacturer. With automatic content recognition of the program (versus just channel), number of demographic info, IP address, etc. All of that was sold to advertisers.*

- (Background:) *Robot vacuum which additionally created a floor plan of the apartment and uploaded it to the manufacturer, which was then re-used for advertising purposes.*
- (Background:) *Smartphone apps (potentially even smart devices) use the microphone to recognise which TV channels or even movies in theatres you are watching. Organisation that distributes this technology is called Alphonso. With the data sold again to advertisers.*
- (Background:) *Smart TVs with voice control functionality may unexpectedly also record normal conversations, and transmit them (unencrypted) to the Internet. This has already been exploited by attackers. Even smart toys had in their terms and conditions that they evaluate and analyse the voice inputs on their server for other purposes and research.*

Insight and Abstraction

- (If **existing tool** in use:) When you think about the existing information of your tool: Which **additional details** would like to know about your devices' communication behaviour in order to feel informed?
    - Especially with regard to unwanted and potentially unknown data transfers.
    - (Examples for technical info:) *Time-based visualisation of the communication, IP addresses, local device names, contacted websites (domain names), involved organisations, amount of transferred data, protocols, encryption*
- (If currently **no tools**:) When you think of unwanted and potentially unknown data transfers of your devices: Which **information** would you like to have about your devices in order to feel informed?
- (Alternative wording:) What would you like to know in order to detect unwanted behaviour of your devices?
- (Similar:) Which aspects of a device's communication would you like to know in order to gain more **trust** in your devices?
- **Contacted servers** are often identified by their (domain) names (google.at, youtube.com): What other information about servers that your devices contact would you be interested in?
    - Especially if the name is perhaps cryptic or unknown to you.
    - (Examples:) *textual description, geographic info, associated organisation, purpose of the organisation / domain* (Cues: functionality, advertiser, tracker)

- (If associated organisations were not yet mentioned:) As an idea: For cryptic server names, the **organisation** behind them may provide additional information. Would you find this worth knowing?
- Are you concerned that your devices are transmitting data to unwanted organisations?
    - (If yes:) Can you give examples of such organisations?
    - (If yes:) Are there generally organisations you don't want to transfer any data to? (Or:) Types of organisations you want to avoid?
    - (Note: Classification by **purpose** like functionality, advertising, tracking, etc.)
- When you think about your devices' communication behaviour: Which **further details** would you like to know about the communication / data transfer?
    - Besides the name of the contacted servers / organisations
    - (Examples:) *Amount of transferred data, time of day, used protocols, encrypted or not*
- (As a transition to the **drawing task**:) How could you imagine a **visualisation** of the information important to you?
    - Since it's often difficult to verbalise visual ideas, I would like to ask you to create a **rough sketch** of your first ideas.

Drawing Task

It's helpful to **think aloud**. It's not about creating a perfect solution or drawing a finished user interface, but primarily about **collecting ideas**. I will gladly help you put some abstract ideas into more concrete concepts. All ideas are allowed, no matter how unrealistic they may seem.

Perhaps as suggestions during the drawing:

- (If sketch contains a lot of detailed information:) How do you imagine a **central overview** of the most important information about your devices. In terms of their communication behaviour.
- It would also be interesting to know how to **group** or subdivide information in a usable way in order to get an overview even with large amounts of information.

*Also ask whether they can imagine **alternative visualisations**. Especially if they used a similar visualisation to existing tools.*

Follow Up Questions

- (Similar to the question about grouping:) For a good insight into devices' communication behaviour, an analysis over a **longer period of time** may be necessary. Taking into account the constantly growing amount of information: Would you **adjust** your visualisation to avoid being overwhelmed with too much information?
    - (If yes:) How?
    - (Examples:) *Filter (show, hide), subdivision / grouping / hierarchy (purpose, organisations)*
- (Related question:) If you imagine checking the analysis of your devices again after a few weeks: What information would you want to **highlight**?
    - (Or:) Can you think of some additional information in order to keep the visualisation relevant in the **long term**? Since one doesn't usually look at it daily over a longer period.
    - (Examples:) *Trends, time-based visualisation*
- (If a time-based visualisation has not yet been discussed:) If you think of a longer analysis period: Is a **time-based visualisation** of interest to you?
    - (If yes:) How could this look like? What information would it contain?
    - (Example:) *How much data was transferred from a device, at what time, where*
    - (Note:) *Mysterious data transmission every day at 1am in the night.*
    - (Example:) *Show data filtered by the last X hours, days, weeks*
    - (Example:) *A graph of the transmission over time*

Interaction and Further Actions

- (Asked again:) If unwanted communication behaviour has been detected, which **actions** should a tool offer as support?
    - (Or:) What **control** over the communication behaviour do you desire?
    - (Example:) *Blocking individual connections*
    - (Example:) *General notifications*
    - (Example:) *Automatic blocking of connections to ad trackers*
- (In the context of the created sketch:) Which **interactions** did you have in mind for your design?

    - (Examples:) *Show or hide pages / organisations / devices; manual grouping; changing the name of organisations*

- (Optional, maybe concluding question:) Is the **content** of the transferred data also relevant?
    - (If yes:) In which situations and in which not?

Conclusion

- Are there any more questions you **expected** me to ask?
- Are there any other stories about **problems with smart devices** that you would like to tell me?
- *Say thank you for their time, and ask if they are interested in participating in the* ***case study***
- *Short summary of my findings from the current interview and more detailed information about the goals of my research*