Privacy Controls for Always-Listening Devices

Nathan Malkin

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2022-249
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2022/EECS-2022-249.html

December 1, 2022

Privacy Controls for Always-Listening Devices

by

Nathan Malkin

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor David Wagner, Co-chair
Doctor Serge Egelman, Co-chair
Professor Vern Paxson
Associate Professor Raluca Ada Popa
Assistant Professor Florian Schaub

Summer 2021

Privacy Controls for Always-Listening Devices

Abstract

Privacy Controls for Always-Listening Devices

by

Nathan Malkin

Doctor of Philosophy in Computer Science

University of California, Berkeley

Professor David Wagner, Co-chair

Doctor Serge Egelman, Co-chair

Intelligent voice assistants and other microphone-enabled Internet of Things devices are increasingly popular but may pose grave privacy risks. As more technologies take advantage of continuous audio access, new products may emerge that are passively listening and actively analyzing all conversations they hear, instead of only processing commands that follow their wake-words. The research in this dissertation investigates how to develop a privacy permission system for these devices. It takes a user-centered perspective on this problem and uses empirical methods to study the research questions that can guide the design of the privacy controls:

1. What are people's expectations about what information needs to be protected and in what context?
2. Which privacy-enhancing techniques could be feasibly applied to limit the devices' listening? How do people perceive their trade-offs and acceptability?
3. Which interfaces and affordances would allow users to express their privacy preferences and explore the implications of their choices?

This dissertation explores these questions through a combination of surveys, user studies, and prototype evaluations. Major conclusions include:

1. People exhibit nuanced and heterogeneous preferences, notably in relation to other members of their households, and are especially wary of undisclosed data flows to third parties. They are most protective of financial data and other information that can cause them harm.

2. Block-listing and filtering approaches may be most feasible to implement. In combination with existing techniques and privacy-friendly design choices, they can address immediate user requirements. However, more complex privacy needs must be addressed with content-based controls, which require additional research in privacy and natural language understanding.

3. People appreciate the control install-time and runtime permissions provide over their own data. However, both have challenges with their user experience. Transparency-based approaches may be comparatively frictionless.

Contents

List of Figures

List of Tables

Acknowledgments

When I happen upon a thesis of any kind, the acknowledgments are the first thing I read; this is my favorite section. Acknowledgments uncloak the carefully crafted character of the lone scholar in the proverbial ivory tower. They show that behind the polished prose and tedious typesetting is a human who wouldn't have been able to do this alone. I know I couldn't have.

* * *

I'd like to thank the academy—by which I mean the educators, both formal and informal, who have nurtured my curiosity.

First and foremost, I am grateful to my advisors, David and Serge, who gave me the freedom to pursue my interests and offered advice when I inevitably got lost.

My committee members—Florian, Vern, and Raluca—provided valuable feedback on this dissertation.

I also want to express my appreciation for my undergraduate advisors, David and Shriram, as well as Brown's UTA program. Without these positive experiences, I wouldn't have embarked on something as foolhardy as a PhD.

My research wouldn't have been possible without my coauthors and collaborators. Primal, Allen, Joe, and Gary contributed to the studies in this dissertation. At other points during my PhD, I was fortunate to collaborate with Alisa, Julia, Maritza, Frank, Lisa, Marshini, Tomasz, Madiha, Heather, Arunesh, Eyal, Marian, Alex, Linda, David, Ganesh, Rob, Adrienne, Giselle, Alex, Karen, and Alison.

A number of students at Berkeley and beyond also participated in my research, and I appreciate their time and efforts: Allen, Arjun, Ashkan, Bhuvan, Bryant, Dylan, Gary, Hrithik, Jiwon, Joe, Katrina, Nikhil, Prakash, Shiven, Varun, and Vikranth.

I would like to acknowledge my funders—federal, corporate, and non-profit—who enabled the genuine academic freedom that I had during my PhD. I am grateful, in particular, to the Center for Long-Term Cybersecurity and its staff for their support. Thank you also to the anonymous donor who endowed the Cal Cybersecurity Fellowship.

Most of the work in this dissertation is "human subjects research." Behind this clinical term are thousands of people who took time out of their lives to answer questions and perform tasks, even though—as our consent forms emphatically noted—"there is no direct benefit to you by participating in the research."

University bureaucracies can be relentlessly byzantine, so I'm grateful to the staff at Berkeley and other institutions who enabled me to sail smoothly through. Thank you Audrey, Jean, Lena, and especially Angie for going above and beyond.

I've greatly enjoyed the company of, and the opportunity to learn from, the grad students of the EECS department, including An, Becky, Bill, Chawin, Chris, David, Jethro, Linda, Lynn, Michael, Mitar, Nick, Norman, Paul, Riyaz, Thurston, Tobi, Warren, Weikeng, and Yifan.

I value the usable security group at ICSI and Berkeley as both colleagues and friends. Thank you to Alisa, Conor, Irwin, Julia, Nikita, Noura, Primal, as well as our ICSI visitors:

Arunesh, Joel, Joshua, and Rakib. I would also be remiss if I did not acknowledge the influence on my work of one of our lab's longest-serving members, Refjohürs Lykkewe.

The 721 Soda office, as well as its extended members, have made this PhD at least an order of magnitude more fun that it would've otherwise been. I'm very happy to have met Austin, Frank, Grant, Pratyush, Richard, Rishabh, and Wenting, as well as Aparna, Eric, Ian, Patrick, and Shirin. Thank you also to my friends outside the academic realm, including Hari, Mattathias, and Mike.

I feel secure knowing that, whatever I set out to do, my family has always been, and will be, behind me. I treasure the time we spend together and value the messages, calls, and support I receive when we're apart. A colossal thank you to Clara, who encouraged me to pursue the PhD and bore the brunt of that decision. Thank you for being there all this time. △

I know there are many others without whom I would not be where I am today, and I'm incredibly grateful for their influence on my life.

To everyone—thank you!

* * *

This dissertation is dedicated to my grandparents. They survived adversity that I can scarcely comprehend, yet they remained full of of humor and humanity, curiosity and kindness. That they were able to do so and achieve so much is a reminder and an inspiration.

Figure 0.0: I'd like to further acknowledge Charlie for both impeding and enabling progress on this dissertation.

The telescreen received and transmitted simultaneously. Any sound that Winston made, above the level of a very low whisper, would be picked up by it, moreover, so long as he remained within the field of vision which the metal plaque commanded, he could be seen as well as heard. There was of course no way of knowing whether you were being watched at any given moment. How often, or on what system, the Thought Police plugged in on any individual wire was guesswork. It was even conceivable that they watched everybody all the time. But at any rate they could plug in your wire whenever they wanted to. You had to live — did live, from habit that became instinct — in the assumption that every sound you made was overheard, and, except in darkness, every movement scrutinized.

George Orwell, *Nineteen Eighty-Four*

Chapter 1

Introduction

This introduction helps answer the following questions: which specific "always-listening devices" are we studying? Why should we study privacy controls for them? What is the roadmap that the research in this dissertation takes for designing the controls?

> Numerous mechanical devices threaten to make good the prediction that "what is whispered in the closet shall be proclaimed from the house-tops."
>
> Samuel Warren and Louis Brandeis, *The Right to Privacy*

The research in this dissertation is aimed at illuminating the design space of privacy controls for always-listening devices. But what exactly constitutes an always-listening device and why might it need privacy controls?

1.1 The emergence of always-listening devices

The Internet of Things is the future. For many years now, the vision of a world where more and more devices are connected to the Internet has been evangelized and extolled, forecasted and feared. But, to a large extent, the Internet of Things is already here. We talk to smart assistants on our smart phones in our smart homes. There are billions of Internet-connected devices all around the world. They carry considerable conveniences; however, they also produce perplexing privacy problems.

Intelligent voice assistants form one category of existing technologies that exemplifies these tradeoffs. These assistants—for example Amazon's Alexa, Apple's Siri, Google As-

sistant, and Microsoft's Cortana—respond to users' voice commands, such as playing music, looking up information, or activating a smart-home device. They are available in most smartphones, and their voice-enabled interfaces have also been added to conventional gadgets like televisions, microwaves [222], and even toilets [36]. However, their most ubiquitous form factor is the smart speaker. Popular models include the Amazon Echo, Google Home, and Apple HomePod. In just a few years, smart speakers have reached a high percentage of households in the United States [85, 100, 167] and hundreds of millions of users around the world [31, 112].

Regardless of their form factor, voice assistants operate by listening for "wake-words" (such as "Hey Siri" or "Ok Google"), then recording and analyzing any audio that follows it, in order to extract (and act on) the user's instructions. In effect, this makes these devices always-listening, since their microphone is on all the time. However, today, the scope of the listening is quite narrow: detecting the device's wake-word. This task is therefore typically performed offline, with extraneous audio ignored or discarded.

Even the current setup, with users explicitly triggering the device, presents privacy difficulties. Most people do not understand when a smart device is listening and where it is sending data. Some may not even realize that their recordings are stored in the cloud—though law enforcement certainly does, and has requested the collected data as part of investigations [144]. While the possibility of mass surveillance, as imagined in George Orwell's *Nineteen Eighty-Four*, exists as a stark fear for some, most current threats remain less totalitarian in nature. Advertisers have sought to exploit the insights these devices offer into shoppers' lives. Patent filings from Amazon and Google have described designs for using data they collect from smart speakers for targeted advertising [194].

Patent filings provide clues to other features that we may expect to one day see in our assistants. One notable assumption is that the devices will be constantly listening and analyzing data [64, 170]: for example, the assistants may be expected to detect a fire alarm going off, a baby crying, or a child "engaging in mischief" [135]. In fact, some of these features have already been incorporated into the Alexa assistant [232]. More simply, users may want to vary their sentence structure when addressing their devices ("turn on the lights, Siri" instead of "Siri, turn on the lights"). Reverse engineering Google's app has revealed experiments doing away with the standard wake-words in favor of direct invocation [229].

Always-watching devices are readily available. The Google Clips camera (released in 2017) takes pictures continuously and then selects the "best" photos among them [172]. Less sophisticated but more common, in-home cameras are marketed for a variety of purposes: security, watching pets, and surveillance of nannies and other domestic workers. Most are already connected to the Internet, and, with time, they too are likely to incorporate AI-powered or other "smart" features.

Voice interfaces may similarly benefit from an "always-on" capability. Spotify's plans reveal an interest in analyzing voices in order to suggest songs based on people "emotional state, gender, age, or accent" [184]. Fitbit, the maker of wearable fitness trackers, has been experimenting with making their devices listen to "ambient noise including your potential snoring" [134]. Amazon has released their own fitness tracker, the Halo, that aims to determine its wearer's mood by continuously listening to the tone of their voice [19, 101].

1.2 The need for novel privacy controls

Smart speakers have the distinction of being one of the most privacy-sensitive Internet of Things (IoT) devices. All smart-home devices have the potential to reveal information about their owners' habits, but research has repeatedly shown that people find two data types most sensitive: audio from conversations and video from inside the home [140, 124, 151, 163, 75]. As these devices' features expand, it is highly likely that more people will welcome them into their homes without full knowledge of potential privacy ramifications.

What does it mean for consumers' privacy if a device is always listening? Rather than waiting for a wake-word to start capturing the audio surrounding the device, they will be "passively listening" to everything until they hear content relevant to their functionality. This calls for a shift in how we think about privacy on these devices—and in our homes. Previously, we opted in to interactions with our assistants, addressing them by name. Soon, the default will be for our speech to always be collected, and we need to think about how to opt out (i.e., block) the devices from gaining access to certain speech. But with the trigger word gone, users will have even less insight into when these devices are capturing or processing audio, significantly increasing the chances of a device capturing audio in unexpected—and potentially privacy-invasive—circumstances.

Lack of transparency will not be the only privacy issue for future passive listening devices. Depending on where in the home they are located, they can listen in on different conversations with varying degrees of sensitivity. Based on what they hear, they will be able to infer highly private and intimate details about our lives. This further raises difficult questions of whether this data is stored and who has the opportunity to review it. Compounding the challenges, multiple people could be involved in a conversation, each with different privacy expectations and preferences. Many, due to their status in the household, may lack any administrative access or accounts with the service. Some, like visitors, might not even know about the device's presence. Meanwhile, as assistant-enabled devices become smaller and more ubiquitous, their form factor and hardware limitations will make privacy indicators and feedback cues less feasible.

The picture is further complicated by the ecosystem that will surround these devices. We envision the ecosystem following the path of smartphones, where manufacturers will

serve as a mediator (like the Android and iOS operating systems do) for a multitude of third-party applications. These apps will provide narrowly defined functionality, such as calendaring, ride sharing, etc. Some voice assistants already support third-party functionality (e.g., "skills" for Alexa) and are actively promoting their platform to developers.

What will the privacy protections be when these devices shift to passive listening and start sharing audio with third parties? Will the third parties also get full access to all audio from within our homes? Because of liability and reputational concerns, platforms themselves will likely want to impose restrictions on the data third parties collect. The question—and challenge—is, how? How can a passively-listening device identify when it should or should not be listening? How can a person specify which conversations are fair game for an app and which are private? Can a conversation be appropriate for one app while being inappropriate for another?

Answering these questions requires new approaches for both privacy and security. While existing paradigms—from capability-inspired permission systems to access control—present a full spectrum of options, they are by and large inadequate for the problems posed by passively-listening devices. Granting a "microphone permission" to Siri or Alexa does nothing to limit what it might hear and how it will use the data. Instead, we argue that privacy controls must make use of the content of communication rather than metadata about it. These would expand on the capabilities of existing permission systems, which do not examine data directly, but rely on metadata about it to determine if flows are appropriate. However, these methods may have only limited applicability to speech controls, since conversations may have identical attributes, differing only in their content.

This brings us to the central *research question* for this dissertation: **what are the appropriate privacy controls for an always-listening device?** Naturally, this raises new questions of its own: what exactly do we mean by "always-listening device?" And what would make for an appropriate privacy control? These questions are addressed in the next two sections.

1.3 Envisioning a passively listening assistant

To begin developing privacy controls for always-listening devices, we need to first clarify *which* always-listening devices we are talking about. As we have seen above, continuous listening may be built into a variety of consumer gadgets, from fitness trackers to household appliances. While some controls may be appropriate for a broad range of products, it will be easier to make progress if we are more narrowly focused on one type of device. For this purpose, we have selected a *passively listening voice assistant* as the object of our research.

In our vision, we are dealing with what is essentially today's smart speaker: an audio-

and microphone-enabled device with an assistant on-board. This assistant, we imagine, will be endowed wth *passive listening capabilities*: it will continuously listen and analyze all audio it hears, rather than just searching for wake-word invocations. In return, it will be able to offer answers and suggestions based on users' conversations, even those not directed at it. For example, it may be able to react to a discussion about the weather with relevant meteorological facts, or a conversation about cooking with relevant advice or tips. We will consider the assistant's capabilities in greater detail later, in §5.4.

We imagine that any passive capabilities will coexist with present-day trigger-based functionality (i.e., users will still be able to directly ask the assistant questions). However, for the purposes of this dissertation, we will focus primarily on the passive listening.

Another assumption we will make is that the assistant device will have a screen for displaying information. This is consistent with the existing trend in voice assistants, where newer models are offered with a screen, for example the Amazon Echo Show, Google Nest Display, and Facebook Portal. This assumption opens up a visual channel for the device's communication with the user: it will be able to display its suggestions in an ambient manner (as opposed to interrupting conversations with the assistant's voice), and the screen could also be used for showing or configuring any privacy controls we come up with. As we begin developing the controls, we will need to make some additional assumptions; we will discuss them in §5.2.

Why study a hypothetical product, when existing devices are also always listening and already suffer from a variety of privacy issues? We see this project as preparing for the future. Given the trends discussed above, we believe passive listening assistants are, quite likely, where the technology is headed. By studying them now, we hope to have some solutions ready when they become available. And any of our findings will also likely be relevant and retroactively applicable to existing devices too.

1.4 Research overview

This dissertation begins to explore the design space for possible solutions to the problem of privacy for always-listening devices. After a survey of related work (Chapter 2), it establishes an operational definition of privacy, based on the theory of contextual integrity (Chapter 3).

Before considering any designs, we need to understand the privacy preferences and expectations of potential users. Our approach to this was to focus on users of current smart speakers (Chapter 4), since that is the closest existing technology to passive listening, and because their users are also most likely to become early adopters of future assistants. Most relevant to passive listening, we found concern about overall data retention and differential treatment of different household members.

After collecting initial expectations, the next step is to begin the design process with an in-depth characterization of the design space (Chapter 5). Because these devices are hypothetical—a challenge we will come back to throughout this dissertation—this involves imagining how passive listening applications might work and inventing examples of their functionality, in order to create a clear picture of the problem we are trying to solve. We then enumerate several potential approaches for ensuring end-users' privacy when interacting with passive listening apps and discuss the trade-offs of these approaches. We also examine how the different approaches can be evaluated and compared.

This analysis also creates a roadmap for the subsequent research: the approaches to consider and the methods to evaluate them. Doing so exhaustively is prohibitive, if aspirational, and so the remaining chapters experiment with different approaches and methodologies, in order to shed light on several potential privacy-enhancing designs.

Chapter 6 compares several of the proposed approaches based on their comfort and acceptability to end-users, aiming to answer the question, which privacy controls would make users more willing to use passive listening devices? We found evidence that stronger privacy controls make always-listening devices more acceptable, but also observed that most people pass over permissions when installing apps.

Chapter 7 chooses one particular privacy architecture and performs a user study (utilizing a Wizard of Oz simulation of the passive voice assistant) with a focus on understanding how users would react to permission requests that happen at runtime. The reactions were mixed: people appreciated having control over the assistant's actions, but were annoyed by frequent interruptions.

Finally, Chapter 8 focuses on another approach—auditing—and explores how well people would perform this task if the only privacy protection offered was transparency into the passive voice assistant's listening decisions. This approach appeared promising for detecting significant misbehavior by apps, though likely insufficient if it were the sole privacy control.

Together, the findings in this dissertation suggest that, while a single control may not be able to prevent all data leakage, a host of design choices can address potential users' privacy expectations. Therefore, the hope is that, taken in concert, the work in this dissertation represents a first step towards a future where—though our devices might inevitably be always listening to us—they are able to respect our privacy wishes in the process.

Chapter 2

Related Work

The research in this dissertation lies at the intersection of several different research directions. This chapter surveys some of these fields' most relevant work.

2.1 Continuous listening and proactive services

This dissertation focuses on intelligent voice assistants with passive listening functionality. While IVAs are quite popular, consumers currently use them for relatively simple tasks, such as playing music, performing searches, and controlling IoT devices [14]. However, researchers have proposed and prototyped much more advanced services, which can offer assistance proactively but require continuous listening. Examples of these include the work by Kilgour et al. [111], who developed Ambient Spotlight to automatically find documents relevant to the current meeting. Carrascal et al. [42] parsed phone calls to help surface important details from them. Shi et al. [192] created IdeaWall, which continuously analyzed conversations, extracting essential information and augmenting it with results from web searches. Brown et al. [35] and McGregor et al. [142] worked on offering proactive actions based on conversations in business meetings. Andolina et al. [15] prototyped proactive search support in conversations. Wei et al. [220] proposed using proactive smart speakers for chronic disease management by finding opportune moments to engage with patients. They then built a prototype of this system and used the Experience Sampling Method to study the best times and contexts for interventions to take place [221]. Völkel et al. [215] studied dialogues people imagined having with perfect voice assistants, finding that they are envisioned as proactive and knowledgeable about the user and their background. In our research, we take inspiration from all of these systems by assuming that assistants will eventually be able to perform similar services.

2.2 Privacy threats from smart home devices

As their popularity increases, Internet of Things devices present an increasing threat to the security of the Internet as a whole [93, 180, 77] as well as to the privacy of individual end-users [37, 18, 63]. For example, researchers have found that just a few seconds of ambient audio captured during user interactions can be sufficient for identifying activities users are engaged in [4]. The security and privacy of smart home devices are difficult for consumers to manage [43], and when security and privacy failures happen, they often have a severe impact on end-users and are difficult to remediate [32].

2.3 Privacy concerns about smart home devices

Because smart home devices enter homes as "smarter" versions of already-existing appliances (e.g., TVs, light bulbs, and locks), users are often unaware of the risks they pose [139]. Because of the mismatch between user perceptions and actual behavior, researchers have sought to document users' privacy expectations, in order to understand where gaps might lead to privacy failures and help device designers create systems better aligned with people's preferences. For example, Zeng et al. interviewed 15 smart home administrators about their security and privacy attitudes, finding gaps in threat models but limited concern [230]. Zheng et al. similarly conducted interviews with smart home owners, focusing on how they made privacy-related decisions [231]; they found a high level of trust in devices' manufacturers, and few attempts to ascertain their privacy claims and data protection practices. Most members of the household in a smart home are "inhabitants" [146], "bystanders" [228], or "passenger users" [115] who do not administer their home's devices; these multi-user environments lead to tensions [86]. Other researchers have focused on collecting more normative judgments, for example Naeini et al. [154] and Apthorpe et al. [17], who investigated which IoT data flows users find acceptable.

2.4 Privacy concerns surrounding intelligent voice assistants

In addition to privacy expectations for IoT devices in general, researchers have studied privacy concerns specific to intelligent voice assistants [120, 140, 45]. Moorthy and Vu observed that people interact differently with assistants in public versus in private [151, 152], and Cho found that the modality of the interaction affects users' perceptions of the assistant [49], suggesting smart speakers' in-home setting as a unique environment.

A number of studies have examined users' concerns about smart speakers potentially violating their privacy. Lau et al. [122] compared the differing concerns by smart speakers

users and non-users by conducting a diary study with 17 smart speaker users, and then interviewing them and a further 17 non-users. Huang et al. [104] interviewed users about risks within their household and external to it. Abdi et al. [1] identified mistakes in users' mental models, especially as they relate to third-party skills. Major et al. [138] found that users struggle at distinguishing third-party skills from first-party features. While these studies all focused on currently available products, In co-authored work, Tabassum et al. [200] surveyed people's perceptions of always-listening devices, such as those our research focuses on, finding that people were interested in the services the novel devices could provide but had reservations about their privacy implications.

2.5 Attacks on smart speakers and skills

Our research assumes that many of the always-listening services hypothesized above will be offered as add-ons created by third-party developers, due to the ecosystems of skills that have already been established around IVAs. (This assumption is discussed in greater detail in §5.2.) Academic and industry researchers have already discovered vulnerabilities in these ecosystems [129, 109, 173, 51, 128, 48, 50]. Vaidya et al. [211] examined whether gaps between how humans and machines interpret speech could lead to security vulnerabilities. Kumar et al. [118] described "skill-squatting" attacks on smart speakers, in which users are tricked into triggering malicious skills, which are given names that sound similar to legitimate skills, introducing the possibility that their invocations are misinterpreted. Mitev et al. [148] developed a skill-based MITM attack on smart speakers. Cheng et al. [48] found that malicious skills could pass Amazon's skill certification process.

2.6 Inappropriate listening by smart speakers

In response to the security issues and privacy concerns described above, researchers developed several techniques for detecting when smart speakers listen when they are not supposed to. Pan et al. [164] searched for third-party apps for Android smartphones exfiltrating audio and video data. Dubois et al. [65] and Schönherr et al. [186] identified instances of accidental activation of smart speakers by playing hours of audio from popular TV shows and other recordings to the devices and seeing if they activated.

Microphone blockers for smart speakers

The threat of accidental activations and concerns about voice assistants spying on their users have led to the development of several technologies designed to prevent smart speakers from listening when they are not supposed to. Tiefenau et al. [203] prototyped a "Privacy Hat" that can be placed on top of the speaker as a more tangible and noticeable way of invoking its mute feature. Chandrasekaran et al. [44] designed two separate interventions: one that cut off power to the smart speaker and another that targeted its

microphones with obfuscation. Other interventions tend to fall into one of those two categories and are also commercially available [155]. Chen et al. [47] developed an ultrasonic jammer for the smart speakers' microphones, which could be worn on one's wrist. Liu et al. [133] investigated jamming using personalized "babble noise" to obscure speech from both automated and human attackers.

2.7 Searching for risky voice apps

Researchers have also worked on identifying potentially dangerous skills in current Alexa and Google voice assistant stores. Shezan et al. [191] created a list of 58 sensitive keywords, then searched for them in voice commands listed as examples in descriptions of Alexa and Google Home skills. The researchers also looked for undocumented voice commands by selecting a sample of sensitive voice commands and seeing whether 50 randomly chosen skills, run in a simulator, responded to these invocations. Guo et al. [95] also extracted sample queries from skill descriptions and fed them into skills running in a simulator. After receiving the responses, they then tried to answer the skills' questions in an automated way, such as by identifying and responding to yes/no questions and drawing on synthetic personas to answer demographic questions. After constructing this corpus of simulated conversations, the researchers then analyzed them for "words related to privacy," uncovering over one thousand skills that requested private information. This work is relevant to our research, because uncovering and blocking sensitive interactions may be one form of privacy controls.

2.8 Longitudinal privacy management

One way in which always-listening devices present a privacy threat is their collection and retention of user data. In considering the retention of smart speaker users' recordings (a particular focus of Chapter 4) our work also builds on research on longitudinal privacy management, which has shown a strong demand for the deletion of old data from users of social media [21, 149] and other online web applications, such as cloud storage [110].

2.9 Crowdsourcing privacy and security evaluations

Some system designs may choose to involve their users directly in identifying privacy threats. (We study this possibility in depth in Chapter 8.) In essence, this is a form of crowdsourcing. Bug bounty programs are one very common example of crowdsourcing security evaluations; a number of studies have investigated their effectiveness [78, 137, 217]. In bug bounty programs, the "crowd" is made up of experts; comparatively less studied is the question of how users without specialized knowledge or training can be useful in discovering security and privacy problems. Agarwal et al. [6] crowdsourced

privacy decisions from iOS users to inform a recommendation engine. Kong et al. [114] mined user reviews of smartphone apps to understand the apps' security-relevant behaviors. Similarly, Tao et al. [201] extracted sentences about security issues from mobile app reviews. Hatamian et al. [98] mined user reviews and categorized the reports of privacy threats and behaviors posed by the apps. Wang et al. [218] also mined user reviews for privacy information, focusing specifically on the apps' permission requests, and found that user reviews were a better predictor than the descriptions supplied by the apps themselves. Eiband et al. [69] examined reviews of smartphones apps for reports of problems users experienced. Nguyen et al. [156] studied the relationship between end-user reviews and security- and privacy-related changes in apps. Völkel et al. [216] pursued a slightly different goal: rather than evaluating AIs, they investigated whether people can prevent a chatbot from profiling them by pretending to have different personality traits. The researchers found that people are modestly successful at this task but regarded it as exhausting

2.10 Explainable machine learning in HCI

Intelligent voice assistants rely heavily on machine learning, but decisions by ML models are notoriously difficult for end-users (and even professionals) to understand [227, 99], resulting in reduced trust of ML-based technologies [69, 147]. Our efforts to give users more control and transparency over these systems are therefore connected to the work on explaining their behavior. Explainable machine learning is a burgeoning subfield of machine learning research [2, 3, 26]. Several works in this area have proposed adding an interactive component to help better explain the behavior of machine learning models [153, 197, 223]. In response to efforts that were not focused on the human-computer interaction aspect of the problem, Abdul et al. [2] have called for an HCI research agenda to make autonomous systems more explainable and accountable. Among the more user-centered research that this sparked, Cai et al. [38] showed that examples can be effective at explaining algorithmic behavior. In their case, the examples were drawings and the algorithm was a sketch recognition classifier.

The research in this dissertation builds on the work above and adds a range of contributions to the literature. We articulate a vision for a new type of voice assistant (Chapters 1 and 5). Another dimension of contributions is a deeper understanding of people's privacy concerns and expectations (Chapters 4 and 7, in particular). We characterize and evaluate potential privacy-preserving techniques (Chapters 5 and 6). We study runtime permissions in a novel context (Chapter 7). Finally, we examine the efficacy of a new type of crowdsourced security evaluation (Chapter 8).

Chapter 3

Defining Privacy for Always-Listening Devices Using Contextual Integrity

This chapter provides a general introduction to the theory of contextual integrity, which forms the theoretical underpinning of this dissertation, supplying its working definition of privacy and motivating its search for more fine-grained—context-specific—controls.

> That the individual shall have full protection in person and in property is a principle as old as the common law; but it has been found necessary from time to time to define anew the exact nature and extent of such protection.
>
> Samuel Warren and Louis Brandeis, *The Right to Privacy*

The aim of the research in this dissertation is to enable users of passively listening assistants to effectively protect their privacy. But how will we know what success looks like? What does it mean for someone to be fully in control of their privacy? This is a rather philosophical question, and digging deep into what privacy actually means is outside the scope of this dissertation. Instead, we adopt and apply the definition of privacy offered by the theory of contextual integrity.

The theory of contextual integrity (CI) provides a definition for privacy and a model for understanding when privacy violations happen. Since its introduction in 2004 [159], it has become popular with privacy researchers in computer science and across the social sciences [23].

Because CI is central to our research approach—and because, at present, it has only been described in relatively long and technical publications—this chapter aims to introduce this theory in a way that is understandable to a less niche audience and explain why and when the theory might be useful. It will argue for why a framework like CI is needed, describe the theory's major ideas, show examples of how it can be used, and discuss some of its limitations.

3.1 What is privacy? In search of a definition

Your privacy is very important. Everyone agrees about this. The UN Charter declares privacy to be a human right. Legislatures around the world pass laws to protect privacy. And companies announce their commitment to privacy in full-page ads (usually after violating it in some way) [143, 97, 22]. But what exactly do we mean when we talk about privacy?

Most people can readily come up with examples of behaviors they would consider privacy-invasive: a peeping tom staring through your windows, a stalker tracing their victim's whereabouts, an uninvited reader perusing a personal diary. But "I know it when I see it" is a cumbersome standard by which to identify privacy violations. Moreover, cultures have different standards [126], and privacy preferences further differ between individuals [68].

When it comes to definitions, dictionaries are a natural place to seek clarity. Merriam-Webster, for example, defines privacy as "the quality or state of being apart from company or observation," or "freedom from unauthorized intrusion." While these definitions help clarify the notion of privacy, it's also apparent that they fail to cover a variety of situations and scenarios, especially when it comes to data and the digital domain. When we make a post on Facebook, are we "apart from company and observation?" Why does some data usage feel creepy even when it is disclosed in terms of use documents? Can it still be considered an "unauthorized intrusion?" Dictionary definitions are too limited to provide insight into these questions and too vague to be operationally useful.

Legal definitions have the potential to be more specific, and recent laws such as the European General Data Protection Regulation (GDPR) [71] and the California Consumer Privacy Act (CCPA) [39] are specifically focused on regulating privacy in the Internet age. But while these laws define terms such as "personal data" and "aggregate consumer information," they lack a succinct definition for the term "privacy," instead codifying it as a series of rights for the data subject (such as the right to erasure and the right to rectification) and responsibilities for the data processor (to meet those rights). Moreover, just because some practice is legal, this does not mean people won't perceive it as a privacy violation, as numerous studies and media scandals attest [182, 67, 116, 166, 103, 226, 106].

One of the main limitations of the definitions we have considered so far is that they are difficult to apply to situations where we need them. Concretely, as researchers and practitioners in computer science, we are often faced with questions about the privacy implications of a system:

- Does (or will) this system violate or infringe on privacy?
- Does a solution preserve privacy or mitigate a privacy violation?

Ideally, a definition of privacy would provide enough insight to help address these questions. Essentially, we're looking for a *model*: something that can explain existing phenomena and be used to predict future outcomes. We need a model for privacy.

The theory of contextual integrity, invented and elaborated by Helen Nissenbaum [159, 160, 158], offers just such a model for privacy. It can help analyze a situation from a privacy perspective and provide insights into how people will react when a new system or technology is introduced. We will now explore this theory in greater detail.

3.2 Contextual integrity: the details

The theory of contextual integrity can be broken down into a few main ideas, each building on the previous ones. The theory is not all-or-nothing: you can adopt and use only some of the ideas while ignoring the rest.[1]

Idea 1: privacy is defined by how information flows

Contextual integrity envisions privacy as the "appropriate flow" of personal information. We'll define what it means for a flow to be appropriate in the next section (Idea 2). But first, we'll explore in a bit more detail why information flow is the most appropriate model for dealing with privacy.

Information flow refers to the transfer of knowledge from one party to the next. For example, when you report your symptoms to a nurse, who shares them with your doctor, who inputs them into a computer, which is then breached by a hacker, who sells the data on the illegal market—each of those points represents nodes through which your personal health information has flowed.

While the notion of information flow is fairly intuitive, CI emphasizes it because there are alternative models of privacy that are also widespread, for example secrecy, data minimization, or leakage [29, 62, 27]. The problem with these models, however, is that they

[1] The structure of this section is adapted from *Contextual Integrity Up and Down the Data Food Chain* [158].

tend to be static and absolute: either something is secret or it isn't. Some information will be "sensitive" or "private"—for example, knowledge about relationships, finances, or health—while everything else will be *not* sensitive, maybe even public.

Contextual integrity, on the other hand, observes that privacy is fluid. As an illustration, consider that we don't hesitate to share gossip with our friends, financial details with our accountant, and health information with our doctor. But something would seem off if our friends started interrogating our tax returns, our accountant demanded a list of our medications, and our doctor insisted we spill the latest gossip.

As this example shows, we can't divide information into "secret" and "not secret," "private" and "public." Nor do friends, doctors, or accountants have "clearance" to access *any* of our sensitive details. In respective *contexts*, we freely share information we would otherwise consider private and off-limits.

Conversely, information that can easily be observed in public (such as a visit to a store and a purchase we make there) can still be considered private when it is taken *out* of the original context—for example, if it's aggregated to create a detailed profile of our movements or shopping habits.

Contextual integrity addresses this problem by considering not only the specific data type, but the information flow as a whole. Who was the intended recipient of the data and what was their role? (We'll discuss the details of the flow a bit later, in Idea 3.) CI postulates that privacy violations happen when there is *inappropriate* information flow. But how do we distinguish appropriate and inappropriate information flows?

Idea 2: information flow is appropriate when it conforms with contextual privacy norms

According to the theory of contextual integrity, information flow is appropriate when it happens according to the norms of a particular informational context. In other words, CI asks, "what are the privacy norms in this specific situation?" If information is shared in a way that runs counter to these entrenched expectations, that flow is inappropriate, i.e., a privacy violation.

In fact, this is precisely how the theory defines privacy:

> *Privacy, defined as CI, is preserved when information flows generated by an action or practice conform to legitimate contextual informational norms; it is violated when they are breached.* [158, p. 224]

While this may appear almost tautological ('a privacy violation happens when you vi-

olate privacy expectations'), this definition draws an important distinction from notions of privacy that are purely procedural, such as the principle of informed consent [57] and other Fair Information Practice Principles (FIPPs) [162]. Under a procedural model of privacy, any information flow might be considered appropriate, as long as certain practices were followed, such as encrypting the data in transit, or getting the user to agree to some terms and conditions.

Informed consent and other FIPPs certainly have their value, but CI says that following them is not sufficient to maintain privacy, just like you're unlikely to achieve security simply by ticking all the boxes on a checklist. Established norms still govern privacy expectations. Privacy concerns won't magically go away just because the user clicked "I accept."

Instead, CI postulates that *norms* are the key determinant for privacy: generally established standards and commonly held expectations about what will happen with the shared information.

Privacy norms can be shared by an entire society or country (for example, being obligated to submit one's fingerprints if arrested on suspicion of a crime) or can be localized to an individual family or workplace (such as a company where all employees know each other's compensation).

Other examples of contextual informational norms include:

- A teacher is expected to share a pupil's grades with the student's parents, and perhaps other teachers, but not anyone else.
- A therapist is expected not to reveal their patient's mental state, *unless* they believe the patient is in danger.
- Citizens are required to report their income to the government, but the government is expected not to make that information public.

In the digital domain, where things change quickly and norms are less established, privacy expectations are often based on people's experiences with more familiar versions of new technologies or their precursors. For example, my prior research has found that people considering the data collection practices of smart TVs apply their expectations for older (non-smart) TVs, rather than smartphones [139].

Norms can also change, sometimes rapidly. Thus, the most reliable way to ascertain a norm is to conduct research into people's attitudes, beliefs, and expectations. Because norms differ between contexts, conducting this research (and, more generally, understanding norms) requires a more precise definition of what constitutes a context.

Idea 3: a contextual norm can be described by (at least) five parameters

So far, we've seen that privacy can be modeled as information flow and argued that the privacy expectations for these flows are governed by norms, which vary according to context. But what exactly constitutes a context?

According to the theory of contextual integrity, a context can be defined by the following parameters:

1. Data type (what sort of information is being shared)
2. Data subject (who the information is about)
3. Sender (who is sharing the data)
4. Recipient (who is getting the data)
5. Transmission principle (the constraints imposed on the flow)

In other words, to figure out the privacy norms at play in a particular situation, you need to identify and consider these five variables.

According to CI, if one of these variables is undefined, the situation is under-specified, and the privacy expectations can't be fully determined. For example, if we don't know what the information is or whom it's about, we can't say how it should be shared. Or if we know those things but we don't know whom it's being shared with, we don't know if privacy violations are occurring.

Data type, subject, sender, and recipient are all fairly self-explanatory; they've already been implicit in our discussion of information flow. The transmission principle parameter is new to CI and therefore requires some explanation.

The transmission principle accounts for the conditions or constraints that restrict information flow or limit it to specific circumstances. For example, according to some norms, a business should share its customers' records with the government only if the authorities have a warrant or court order. Here, the transmission principle is the existence of a warrant: only in its presence does the information flow become appropriate.

Other potential transmission principles include:

- the subject's consent
- the consent of a parent or guardian (usually when the subject is a minor)
- with notice (some sort of advance announcement or disclosure)
- reciprocity ("I'll show you mine if you show me yours")
- subject to legal requirements
- the Chatham House Rule [46] (information can only be re-shared without attribution)

This list is far from exhaustive; there are many other transmission principles.

There may also be other contextual integrity parameters. While CI holds that the five variables (data type, subject, sender, recipient, transmission principle) are generally sufficient for specifying a context, it allows that other factors may influence people's expectations and norms.

One specific example that often comes up is the question of purpose or use (i.e., how some data will be used and to what end). This turns out to be an important factor both from a legal point of view and in people's expectations. For example, smart speaker users share their voice and interaction data with voice assistants, expecting that these will be used to answer queries, provide services, and perhaps improve the devices; however, many would find it unacceptable if this data were used for advertising. This distinction could be represented by a separate "purpose" parameter.

While CI, in its original formulation, lack this purpose/use variable (Nissenbaum, the theory's creator, has written that she is "increasingly persuaded" that it should be included [158, p. 234]), it does provide a framework for addressing this distinction.

CI conceives of actors (subjects, senders, recipients) not as identities (named individuals, companies) but as *roles*. An actor might have different roles; for example, your doctor might happen to be your friend or family member. In that event, privacy norms are determined by their *role* in a particular context: if they receive information in their capacity as healthcare provider, expectations are different than if they had heard the same thing at a family function.

Roles can be used to specify and restrict purpose. Returning to the question of smart speaker users, we can say that they are sharing their data with voice assistant companies *in their role* as voice service providers. If those companies use it for advertising, then they are taking on a different role—that of advertisers—which is outside the expected context.

Regardless of how exactly you choose to model context, it's worth remembering that purpose matters and, more generally, there's more to contextual integrity than just the five parameters.

Idea 4: new norms and flows are evaluated through their context

Using CI's conception of privacy, and a clearer definition for context, we are now able to model existing information flows with respect to the privacy norms that govern them. But what happens if there's a new information flow?

Just because an information flow is new, doesn't mean it's bad: the new flow could still be appropriate. But to be sure, we need to go back and ascertain the norms for this particular flow. What if norms for this specific context don't exist yet? This is especially likely to

happen when we're dealing with new technology.

Consider, for example, a doctor who decides to streamline their workflow by using novel dictation software or saving patient data to an Electronic Health Record (EHR). This is a novel flow, but do we really need to go out and survey people about their expectations to understand whether this was appropriate or not?

The theory of contextual integrity says: not necessarily. CI provides a way to evaluate the *ethical legitimacy* of novel flows. It gives a framework for identifying the strengths and weaknesses of the novel flow, as compared with the status quo.

Specifically, CI suggests three "layers" of analysis:

1. The interests of the affected parties
2. The ethical and political values
3. The contextual functions, purposes, and values

In the case of the EHR, we would first consider the interests of the parties: it will make the doctor's life easier, but will it hurt the patient? Then we'd consider more general ethical priorities, for example values like free speech and freedom of choice—would any of these be hurt? Finally, we'd consider the fundamental purpose of the context—in this case, providing healthcare. Would these goals be undermined by the flow, or any of its consequences? (For example, will patients become less likely to seek care due to concerns about how their data is used?)

After considering all of these factors, we can decide whether the new flow's benefits outweigh the negatives. If so, we may deem it morally legitimate from the perspective of CI.

Clearly, these determinations are still far from objective. However, the framework offered by contextual integrity provides a more structured way of thinking about whether something hurts or enhances privacy.

3.3 Applying contextual integrity

This chapter has argued that contextual integrity offers an effective model and more precise definition for privacy. But how can it be used? Below are four lessons about how contextual integrity can be applied to research and practice.

Lesson 1: think about flows and contexts, not binary categories

It can be tempting to reduce data to binary categories: sensitive or not sensitive, private or public, information that is—or isn't—personally identifiable, etc. Yet, just as anonymous data can often be re-identified, so can "public" data often turn out to be sensitive. All of these binary characterizations fail to acknowledge the context-dependent nature of what people consider private.

Of course, this is not a suggestion to start treating credit card numbers the same way as comments on a blog post. If anything, it's the opposite. For example, if one were to aggregate a person's every public comment and product review into a dossier, and then publish it, that would feel like a privacy violation. Why? Weren't they public already? As contextual integrity explains, it is not enough to consider that the information is public; we need to think about how that information was flowing before, and how that flow changed.

Another illustrative example is the outcry when, in short succession, pretty much every voice assistant was revealed to have been relying on contractors to listen to some user interactions [61, 193, 102, 55, 81]. Many people were upset to discover this new, previously undisclosed, flow, forcing companies to apologize and back-track.

In this situation, companies felt that they were relatively unconstrained by what they could do with the data, since users had already shared it with them. In reality, they were taking interactions that many already saw as ephemeral and generating new data flows on their basis, creating a (mostly invisible) permanent record. The companies consequently learned that the new flows were surprising and unwelcome to people, even though the data technically never left the company and was not shared with "third" parties. These scandals may have been avoided had the companies been thinking in terms of information flows, and also if they had checked how any such new flows aligned with people's expectations.

Lesson 2: check expectations, not checklists

Internet history is replete with services that abused their users' trust and data, then pointed to a line of fine print to justify it: "Can't you see? You agreed to all of this." Courts have been increasingly skeptical of this defense, and contextual integrity explains why it was never satisfactory: what we consider to be a privacy violation is based on our expectations for a particular context, not a set of practices the provider did or didn't follow.

Newer legal frameworks such as GDPR and CCPA are recognizing this and are consequently requiring positive assent with meaningful opt-out options, instead of pro-forma checkboxes that everyone has to click through. Other pro-privacy moves can also be necessary but not sufficient. For example, data minimization, while a positive step, may not,

on its own, be enough to assuage privacy concerns.

Even privacy-enhancing technologies can fall short due to a mismatch in consumer expectations. For example, research found that many users misunderstood web browsers' "private browsing" modes, thinking that their browsing history would be secret from entities such as employers, governments, or Internet Service Providers [84, 96].

As we discussed above, this is not a dismissal of practices like data minimization or informed consent. They are useful tools on the path to privacy—the path, that is, to *following people's expectations* and adhering to norms.

What are those expectations? The easiest way to find out is to just ask! Researchers in a number of academic fields (anthropology, sociology, information science, human-computer interaction) have been studying these questions for years and have developed techniques for discovering expectations [123]. Similar techniques are also used daily by user experience researchers in industry, who are working in large numbers at companies big and small [89].

Lesson 3: account for the complete context

One important thing to remember about expectations is that they are specific to contexts. Therefore, just because something is considered acceptable in one context, doesn't mean it'll be okay in another. For example, social media buttons ("like this! share that!") are considered acceptable on news and lifestyle websites, but raise questions when they appear on health sites. Though the data flows are ostensibly similar, the different contexts mean the expectations are different.

To think through a context and consider ways in which it might differ from more familiar ones, it can help to identify the parameters singled out by contextual integrity: data type, subject, sender, recipient, and transmission principle. If even just one of these variables changes—for example, a new recipient is added or a transmission principle such as reciprocity is lacking—then the entire flow may become inappropriate.

The details of the parameters matter. Returning to the example of human review of voice assistant recordings, we can reason that people may have known their recordings were being sent to the company. However, they likely assumed that their recordings were being processed algorithmically and were never exposed to human beings. Established norms did not account for the listening done by the humans, even if it was done for benign purposes like improving the assistants' performance. In general, research has found that people are wary with their data being perused by humans (as opposed to being processed automatically by machines) and of it being shared with third parties, whether for advertising or other purposes [17, 231].

The details of information flows are relevant to privacy-enhancing technologies as well, because PETs may inadvertently introduce new flows. For example, when web browsers introduced the Do Not Track HTTP header, it was intended for users to signal an opt-out from behavioral advertising; but it actually ended up being used as another signal for fingerprinting browsers and tracking users [157].

Examples like these provide an important reminder that, when introducing changes to a sociotechnical system, we need to verify the contextual integrity of the proposed system:

- Will new information flows be introduced?
- Are existing information flows changing?
- What are the effects of these changes?

The latter question—the consequences of privacy changes—is especially crucial to consider.

Lesson 4: consider the consequences

As we have seen, contextual integrity can help understand privacy implications of new technologies by decomposing novel information flows into their constituent components (data type, subject, etc.). However, the CI framework is also helpful for higher-level reasoning about privacy. This is enabled by the theory's focus on contextual purposes.

Why do we share information with other people? Usually, the information flow serves a specific goal. Data is shared in medical contexts for the purpose of curing patients, in education contexts for the purpose of imparting knowledge to students, and in contexts of the judicial system for the purpose of securing justice. Even casual interactions, like gossip or small talk, serve a specific social purpose.

CI instructs us to consider the consequences to these purposes when analyzing the impact of new flows. This framework can be used, as an illustration, to analyze the concerns surrounding the increased surveillance of students. As part of the pandemic-induced switch to remote learning, students are subject to a variety of new demands on their privacy, from requirements to turn on their webcams and be on video during remote lectures to invasive monitoring of their computers and surroundings as part of remote proctoring [108]. How should we think about the ethical legitimacy of these novel flows?

The rights and interests of students and instructors are a good starting point for this debate. But CI offers an additional question to guide deliberation: do any of these measures enhance student learning? Or do they actually hurt students' education, by drawing their attention away from the subject matter and introducing new stresses? If so, then the new flows are privacy violations and inappropriate.

Similar skepticism should be shown to new flows that endanger the values and purposes of other contexts: health technologies that may make patients reluctant to seek care (e.g., data sharing between healthcare providers and employers), voting methods that may reduce citizens' engagement or increase their distrust in civic affairs (such as certain proposals for online voting). Regardless of the setting or the technology, a full appraisal calls for considering the contextual values.

Ultimately, this perspective is so useful because—just as security is not a primary activity but rather an operational requirement—most people don't care about privacy for its own sake. Privacy enables free speech, creativity, self-expression, experimentation, and other beneficial values and outcomes. When we fight for privacy, we fight for these values too.

3.4 Limitations and opportunities

Contextual integrity isn't the final word in privacy. It has a number of limitations, which are worth knowing about.

As a (rather theoretical) model, CI aims to predict how people will feel about privacy in certain situations; it does not claim that this is how people *think* about privacy or make privacy decisions. You're unlikely to find many people who go into a situation, identify each of the five CI parameters at play, reflect on the context they are operating in, and then arrive at a privacy judgment. Most of the time, our reactions are rooted in our emotions and intuitions.

Furthermore, even if asked to reflect more logically on their decisions, people don't necessarily think about the situation in the same terms as the CI model [82]. And, like any model, CI necessarily simplifies things. As discussed above, there may be other factors that matter, beyond the parameters CI identifies.

Another limitation of CI is its conservativeness. Though it provides a way of adjudicating novel flows based on the moral values at play (Idea 4 above), CI inherently favors entrenched norms. Existing norms can be entrenched for good reasons, but not always. For example, many workplaces have a norm that employees shouldn't share their salaries with each other, but this may have the effect of limiting workers' bargaining power and hurting under-represented minorities. CI provides some tools for reasoning about these disputes, but isn't necessarily the best framework for doing so.

Applying and operationalizing CI remains an ongoing research question. While it has already been used in a number of computer science research projects, there remain questions about how to best use it [23]. There may also be opportunities to incorporate CI more directly into the privacy decision-making of systems.

In time, a new model or an improved definition might come along to extend (or even

replace) the theory of contextual integrity. But CI is already a powerful tool for making sense of and helping ensure privacy. As researchers and practitioners in computer science, everyone would benefit if more of us knew about and made use of contextual integrity.

3.5 Applications to passive listening

The theory of contextual integrity, and the lessons we gleaned from it above, are directly applicable to this dissertation's subject of passively listening voice assistants. The voice assistants' data flows, as modeled by contextual integrity, will be paramount for understanding how people will react to them and their privacy implications. CI suggests that people will accept passive listening, just like they do other novel technologies, as long as the flows are appropriate.

To ensure that the novel assistants' flows are appropriate, we need to understand people's expectations for them. While the devices remain hypothetical, there may be no established norms for them; however, norms *have* been established, to some extent, for today's smart speakers, which will serve as the precursors for more advanced devices. We can therefore study those to understand the expectations that will serve as the baseline for the new technologies. That is the motivation for, and the focus, of Chapter 4.

Another big lesson of contextual integrity is what its definition of privacy implies about what it would take for privacy controls to be effective. Meaningful controls will need to go beyond muting the device, since that cuts off all data flows, not just inappropriate ones. But the theory suggests that we will also need to go beyond identifying and blocking "sensitive" content, because, as this chapter has argued, there is no set criteria for what is considered sensitive. The answer is context-dependent, and a passively listening assistant will be privy to a variety of conversations across many different contexts. A properly tailored privacy control must therefore be able to identify not whether a particular conversation is sensitive, but whether it is contextually relevant to the specific feature the voice assistant wants to invoke. For example, the statement "my head hurts" is relevant and should be accessible to the assistant (or its app) if it is trying to offer medical advice, but is irrelevant and should be off-limits to any apps and features that do not have this as their goal. Achieving this type of narrow targeting may be difficult, but we will consider some strategies for potentially accomplishing this in Chapter 5.

Chapter 4

Privacy Expectations Based on Existing Smart Speakers

We measured people's understanding, preferences, and expectations for existing smart speakers by having them review randomly selected interactions with their voice assistants and then asking them questions about these recordings. We found that many did not know that their recordings were being stored and wanted them deleted, even if they did not consider them especially sensitive.

4.1 Introduction

The overarching goal of the research in this dissertation is to design privacy controls for passively listening voice assistants. A major step towards that goal is ascertaining people's privacy expectations for these devices, since these can guide the design and prioritization of the potential controls. Of course, passive listening devices do not exist yet; we see them evolving, over many years, from today's voice assistants. Moreover, prior research has suggested that privacy expectations for newer devices are rooted in people's understanding of more familiar ones [139]. Therefore, in this first phase of our research, we decided to survey people's expectations for existing assistants, with a special interest in the controls people use, or would want to use. We also wanted to know how well users understood their devices' behavior.

We wanted to focus on existing users since, if they have already adopted today's smart speakers, they may be more likely to use ones in the future. Clearly, this is a somewhat biased participant pool, since people in it have already decided that the benefits they see in microphone-enabled devices outweigh their privacy risks. But are they making informed decisions? Do people understand the privacy consequences and controls offered

to them? In particular, do users know that their interactions are being stored forever by the manufacturers of the devices and that other members of their households may be able to review them at their leisure? Beyond answering these questions, our study examined how users would prefer their interaction data to be used and stored, inquiring as to how long it should be stored, who should have access to it, and what uses are deemed acceptable.

While some surveys may attempt to elicit such preferences abstractly, we felt that we could get more meaningful responses if people shared their preferences with specific instances of their interactions in mind. To achieve this, we developed a novel technique of using a web browser extension to augment our survey by embedding audio recordings of participants' real interactions with their own smart speakers. This allowed us to probe users' data retention preferences based on recordings of them that were currently being stored by the manufacturers of their devices.

Our contributions include findings that:

- Almost half of participants did not know their interactions were being permanently stored
- Most did not know that they could review their past interactions
- On the whole, data currently stored with voice assistants is not considered sensitive
- Yet, many expressed dissatisfaction with the current retention policies: for over half of recordings, participants considered permanent storage unacceptable, despite that being the current default
- Many find the current practice of manufacturers' employees reviewing their interactions unacceptable
- Respondents appeared more uncomfortable with the storage of others' voices, such as their children
- Few reported making use of existing privacy features
- The majority embraced proposals for alternative privacy features, stating that they would adopt automatic deletion of their recordings

All in all, our results suggest that smart speaker owners are not fully informed about the behaviors and privacy features of their devices. Furthermore, while not many participants considered currently-stored data sensitive, there is a clear gap between people's preferences and the smart speakers' current retention defaults. Our study sheds light on these issues, and our hope is that these results will help guide smart speakers to be more respectful of users' privacy preferences.

4.2 Methods

In designing our study, we were guided by a set of research questions about people's beliefs and attitudes about smart speaker privacy:

- Do users understand that their recordings are being stored forever? Are they okay with this? If not, what might be a more agreeable retention policy?
- Are users aware that they can view their interaction history? Do they take advantage of this feature? If so, how do they use it?
- How do multi-user households use the history feature? Do owners review others' recordings? Do they realize their own interactions may be reviewable?
- What other privacy concerns do people have? Do they take any measures to address these?

While it is possible to answer these questions by surveying people's opinions abstractly, we wanted participants to answer our questions while thinking about concrete interactions they have had with their device. Inspired by the experience sampling methodology [121], including recent studies in the usable security domain [177], we chose to present participants with specific past interactions, then follow up with questions about them. To achieve this in a privacy-preserving manner, we built a browser extension to support and distribute the survey.

The browser extension

To have participants reflect on a representative sample of the interactions they had with their smart speakers, we decided to select several recordings at random. We also wanted the survey to be self-guided, remotely administered, and, most importantly, we wanted no direct access to the interactions ourselves. We chose to achieve this by building a browser extension, through which participants filled out our survey. (We limited our study to owners of Amazon and Google devices, as other smart speakers, such as Apple's HomePod, have a much smaller user base [100].)

After participants provided their informed consent, our extension would make a background request to Amazon or Google's servers, retrieving a list of interactions the user had with their smart speaker.[1] The interactions were held in the extension's memory, without being saved to disk or transmitted anywhere. At the point in the survey where it was needed, one of the interactions would be selected at random and shown to the participant.

[1] "Interactions," as we refer to them, consist of two components: the URL of the audio recording on Amazon's or Google's servers and the transcription of the query, as understood by the voice assistant.

Since present-day natural language processing is far from perfect, accidental recordings regularly occur, sometimes with drastic consequences [188]. Voice assistants are sometimes able to detect these, displaying "transcript not available" or "interaction not intended for Alexa." We used text comparison to screen out interactions the assistant already recognized as invalid, in order to only ask participants about those the voice assistant thought were real, since these are the ones likely to cause unexpected behavior. However, some participants still encountered recordings that did not contain speech or only contained the device's wake-word. Fortunately, respondents were able to derive interesting insights even from these recordings.

Since neither Amazon nor Google provide public APIs for accessing interaction history, we reverse-engineered the HTTP requests and API calls made by the web-based interfaces for reviewing one's own interaction history. Since we were making requests from the user's browser, the requests automatically included the participants' cookies[2] (so the extension never had access to users' emails, passwords, or authentication tokens), and the browsers visited the pages exactly as if the user was manually browsing the review interface on their own. Because participants accessed their own data, on their own machines, with their own authorization tokens, our study was in compliance with the devices' terms of service and relevant US laws (e.g., CFAA).

We developed our extension for the Chrome browser, as it is currently the most popular web browser, with over 70% of the market share as of December 2018 [198]. We made our source code[3] publicly available and linked to it from the recruitment and extension installation pages.

The initial version of our extension sampled from a user's complete interaction history to achieve a uniformly random selection; however, a pilot revealed that this resulted in some participants waiting for almost ten minutes while their full history was retrieved. As a result, we cut off download attempts after 90 seconds and continued only with interactions that had been downloaded up to that point. This cut-off affected 25.9% of participants in our study. While this created a slight bias in our data towards newer interactions, our extension was still able to sample from a pool of thousands of interactions for each such participant (median 4,318, minimum 2,338), going back as far as 22 months.

Survey flow

Our survey consisted of several subsections (the complete survey is in Appendix A.2). Once we obtained consent and confirmed eligibility, we began the survey by probing our participants' familiarity with their device's review interface. Were they aware that the

[2]Participants who were logged out were asked to open a new tab and log in before proceeding.

[3]The extension source code is available at
https://github.com/nmalkin/smart-speakers

voice assistant was storing their interactions? Did they know that they were able to view and delete them? Had they ever done so? What were their reasons for doing so?

We next asked about situations where multiple people had interacted with the smart speaker. For those who previously reviewed interactions, did they encounter others' search history? Was this something they had discussed? How would they feel about others reviewing their own interactions?

At this point, we presented participants with a randomly selected interaction. We first asked general questions about the recording. Who was in it? Was the recording accidental? What were they asking their voice assistant (if they were comfortable sharing)? We then asked participants how acceptable it would be for their interactions to be used, under different circumstances, for these uses: quality control, developing new features, advertising, and others. We also asked participants to rate the acceptability of several different data retention policies. The extension then selected another interaction at random, and the questions repeated, for a total of five recordings.

Afterwards, we asked participants how long they thought the voice assistants should store their data, as well as whether they would use hypothetical privacy controls. Finally, we asked participants whether they had previously had any privacy concerns about their smart speakers and whether they had taken (or now planned to take) any actions to protect their privacy. (We avoided any direct mentions of privacy until the end of the survey.) We ended by collecting basic demographics: participants' ages and genders.

The survey consisted of a mix of multiple-choice questions, 5-point Likert acceptability scales ("completely unacceptable" to "completely acceptable"), and open-ended response questions. Open-ended responses were analyzed using standard practices for thematic analysis [34]: two researchers independently identified themes before collaborating on a master codebook; each independently coded every response, and the two then met to agree on the final codes. We computed inter-rater reliability using the metric by Kupper and Hafner[4] [119]; the mean score was 0.795, and individual scores are listed throughout the text.

Recruitment

We recruited participants from Amazon Mechanical Turk, screening subjects to ensure that they were located in the United States and had a 99% task approval rate. Additionally, we required participants to have owned their smart speakers for at least one month and to have interacted with them a minimum of 30 times. Finally, since our survey was only accessible through the browser extension, the study advertisement stated, as an eligibility requirement: "You use (or are willing to install) the Google Chrome browser."

[4]Commonly used measures of inter-rater agreement, such as Cohen's κ, assume assignment of labels to *mutually-exclusive* categories, whereas we allowed multiple labels per response.

The recruitment posting included the complete eligibility requirements, a description of the tasks to be performed, links to the extension and its source code, and the study's consent form. The task advertisement did not mention or even allude to privacy; it invited participants to "a study about smart speakers" that asked "questions about a few specific interactions you've had with your Alexa/Google device."

The survey took 10–20 minutes for those who completed it in a single sitting. Participants were compensated $5.00 for their participation. All procedures in our study, as well as the recruitment posting and consent form, were reviewed and approved by our Institutional Review Board.

4.3 Limitations

Our methods introduce biases which may have had some effect on our final results. Since we recruited participants from Mechanical Turk, our participant pool may be younger and more technologically literate than the average person.

By surveying current owners of smart speakers, we avoid learning about the privacy concerns of people who refrain from using smart speakers due to such concerns. (Work such as Lau et al. [122] help fill this gap.)

We surveyed only the devices' primary users—those who could control the device. Future work should consider the needs and concerns of household members who lack administrative privileges to the device. (For initial exploration of this topic, see Geeng and Roesner [86].)

By asking participants to download and install a browser extension, we may have turned away those who were more privacy-sensitive and therefore less willing to install third-party software. (As one person who *did* participate in the study wrote, it's a *"bigger leap of trust to install this extension than to worry about Google spying on me for no reason."*)

Due to these factors, and our overall sample size, we do not claim that our sample is fully representative of smart speaker users and their concerns. Nonetheless, we hypothesize that our results illuminate trends present in the larger population and that our unique methodology provides insights that may not have been discovered using more traditional surveys.

4.4 Participant overview

We conducted our study during February 2019. We piloted our study with 13 subjects, then ran the main study with 103 participants, for a total of 116 respondents to our sur-

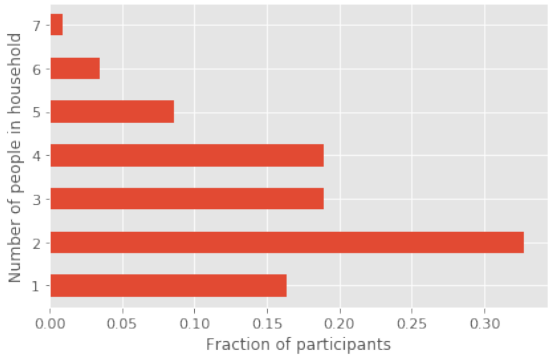


Figure 4.1: Household size among participants.

vey.[5]

Based on the pilot, we made changes to the extension (see Section 4.2) and added two new questions.[6] We made no other changes to the survey, the two studies were conducted back-to-back, and the recruitment procedures were identical; as a result, the main study and pilot were substantially similar, so we report the results as one combined dataset.

Our sample was approximately gender-balanced, with 44.0% self-identifying as female, and the median reported age was 34. Households of 2 or more accounted for 83.6% of all participants, with a median household size of 3 (Figure 4.1).

Device Distribution Approximately two thirds of our participants (69.0%) owned a smart speaker with Amazon Alexa (such as the Echo, Echo Dot, etc.), while the remaining 31% owned a Google Home or one of its variants.[7] These proportions are consistent with consumer surveys, which have found that Amazon holds 70% of the smart speaker market share [100]. There were no significant differences between owners of Alexa and Google devices in their gender (Fisher's exact test,[8] $p = 0.158$), age (independent samples t-test, $p = 0.61$), or the number of interactions they had with their smart speakers (t-test,[9] $p = 0.277$). There were also no statistical differences between the two populations on

[5]Our sample size was motivated by the exploratory nature of the study. Since our hypotheses were not associated with a specific "effect," we did not perform a power analysis.

[6]The questions added after the pilot were the two in Section 4.5 that start with "Suppose the assistant…" We saw during the pilot that participants were interested in automatic deletion, and so wanted to further tease apart some of the factors they brought up.

[7]Respondents who had both Amazon and Google devices were asked to "select the one you use the most," thus the two samples were independent.

[8]Fisher's exact test was used in favor of the chi-squared test because it is more accurate for smaller sample sizes like ours.

[9]When the t-test was used, for age and number of responses, we verified that the data was normally distributed using Q-Q plots, which are included in Appendix A.1.

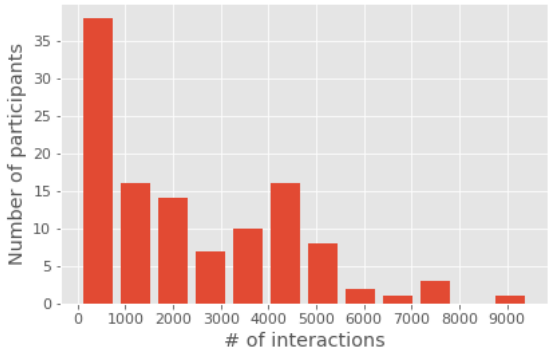


Figure 4.2: Number of interactions obtained per participant.

other questions we tested (see Section 4.5). We therefore report results from the combined population in the remainder of this chapter.

Device Age and Total Number of Interactions Both the mean and the median of the self-reported device age was 14 months. We verified this using the timestamp of the oldest recording obtained from the participant and found that these were largely consistent with the self-reported ages: the median deviation was less than a month (21 days), despite the bias introduced by not downloading the oldest interactions for some participants (see Section 4.2).

The median number of interactions obtained from each participant was 1,850, with a standard deviation of 2,076 (Figure 4.2).

Typical user interactions

To characterize the interactions our participants were reflecting on, we asked them several questions about the recordings they heard. (For more in-depth exploration of typical usage, see Bentley et al. [24]) We first asked who was in the recording (Table 4.1). Over half of interactions were initiated by the respondent, with just under a third coming from other members of the household, including at least 6.75% that were attributed to children.

We next asked respondents to characterize the recording ($IRR = 0.863$). (Subjects could skip this question if they were uncomfortable.) Other than recordings that only contained the wake-word (14.9%), the most common interaction was audio requests (14.0%), where the user wanted to hear a band, genre, podcast, or radio station. Another 10.7% were commands that controlled media, such as changing the volume or rewinding. Users also frequently instructed their voice assistants to control their smart homes (6.57%), tell them the news or weather (4.80%), or set a timer (4.62%).

This is a recording of me.	53.5%
This is a recording of someone else in my household.	32.4%
This is a recording of a guest.	4.3%
This is a recording of noise/gibberish.	2.9%
This is a recording of the TV, music, or other pre-recorded audio.	1.7%
This is a legitimate recording or transcript, but I'm not sure who said it.	1.6%
Other	3.6%

Table 4.1: **Who is (primarily) speaking in this recording?**

Accidental recordings

Voice assistants are only supposed to process interactions after they hear their wake-word, but since this detection is imperfect, accidental recordings may occur. This is one of the major privacy concerns with smart speakers (a fact corroborated by our study, see Section 4.5), and media reports have shed light on incidents where such events had major unintended consequences, such as audio recordings of entire conversations being emailed to random contacts [188]. To better understand this threat, we wanted to know how frequently accidental recordings occur.

Participants reported that 1.72% and 2.93% of all recordings were television/radio/music or just noise, respectively. For all other recordings, we asked: **Did you (or the person speaking) address the assistant, or was the audio recorded by accident?** Respondents said that the speaker was not addressing the device 6.33% of the time. Thus, over 10% of the recordings in our study were unintentional. One participant provided an example of how this may happen: *"I have a friend also named Alexa who comes over, and Amazon Echo thinks we are giving it commands"* (P22).

4.5 Results

In this section, we present the results of our survey.

User perceptions of retention

Prior research suggests that users lack a clear mental model of how voice assistants work [45, 231, 122]. For example, many may be confused about whether processing happens on-device or in the cloud [139].

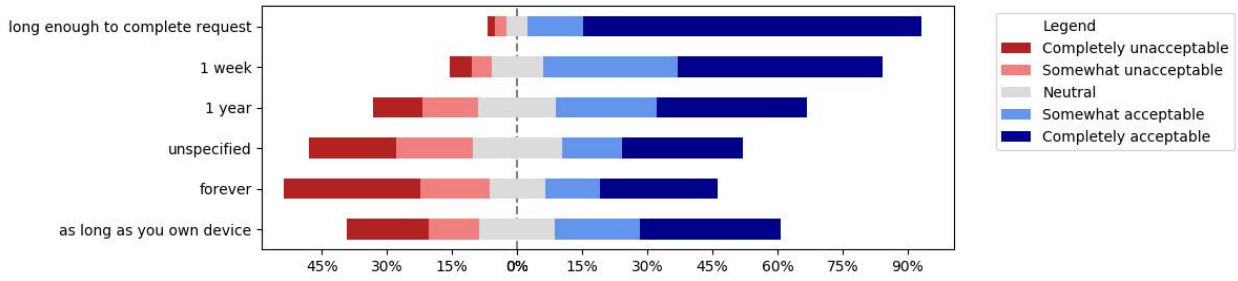


Figure 4.3: **How would you feel if this audio recording were stored for the following periods of time?**

We hypothesized that many people are similarly unsure about what happens to their audio after the assistant answers their query. To test this hypothesis, our survey inquired: **after you ask the assistant a question or say a command, what do you believe happens to the audio of your interaction?** Almost half of respondents, 48.3%, correctly answered that recordings are kept indefinitely by the companies. However, almost as many people incorrectly believed that their audio is only saved temporarily (41.4%) or not at all (4.3%); 6.0% of participants were unsure.

Current data retention policies

Participants shared a range of opinions about the voice assistants' current retention policies. In open-ended responses to various questions in our survey, a number of users expressed unhappiness about the fact that the companies kept the recordings. *"I don't really want any of my recordings stored on Amazon's servers,"* stated one participant (P88). Another wrote, *"I had no idea that the recordings were stored and would prefer that they be deleted. I was kind of shocked to hear my son's voice in the recording"* (P80).

Some were more accepting of the data retention because they saw its benefits and found them worthwhile: *"I think they use the recordings [to] create a voice profile so Alexa gets better at understanding what I say. So [I] will keep all recordings"* (P3). Others seemed to view the device as requiring certain privacy trade-offs and were accepting of these: *"I think [having recordings stored] may help with the technology and we all have to do our part to advance it"* (P111).

To gain more quantitative insights into users' preferences for data retention, we asked participants about each of their past interactions, **how would you feel if this audio recording were stored for the following periods of time?** Participants answered on a 5-point Likert scale from "completely unacceptable" to "completely acceptable" (Figure 4.3).

Participants were much more comfortable with shorter retention periods than longer

Deletion reason	% respondents	% recordings
No need/reason to keep it[11]	34.6%	23.9%
Don't want anything stored	25.0%	35.2%
Not intended for assistant	15.3%	6.3%
Kids	13.5%	8.5%
Not useful to company	9.61%	7.74%
Guests	7.68%	4.23%

Table 4.2: Common deletion reasons, as percentage of respondents who deleted at least one recording (column 1) and percentage of responses marked for deletion (column 2); $IRR = 0.704$.

ones. For 90.8% of recordings they were presented with, participants stated that it would be acceptable for the companies to keep them for one week; that number was only 57.7% for a retention period of one year. Participants were most unhappy with recordings being stored forever; they rated this completely or somewhat unacceptable for 47.4% of recordings.

Consistent with these findings, when we asked **given the option, would you delete this specific recording from <*Company*>'s[10] servers?** 44.8% of participants said they would delete at least one recording (Figure 4.4). However, not all recordings were judged to be a privacy risk, so only 25.7% of the interactions shown were marked for deletion.

We also asked respondents how they arrived at this decision. Given that most people chose not to delete their interactions, ambivalence was common: *"It wasn't anything important [so] I don't care if it's saved or not"* (P84). Other people explicitly considered their interactions from a privacy perspective—and found the value to be low: *"It contains nothing that is a threat to my privacy or identity so I am not especially worried about how it is used"* (P112). Or, more plainly, *"There was nothing that needed to be hid[den] from anyone"* (P99).

Others felt that they needed to keep sharing their information with Amazon or Google to keep the device performing well: *"I noticed that the Dots'/Alexa's performance seems to suffer when I delete basic recordings"* (P18). Some used this specifically as the reason for keeping the recordings—*"if it helps Google get better then no reason to delete it"* (P86)—and would delete recordings that did not fit this use case: *"Because if the information is being used to improve my experience, this is not helpful for that"* (P33).

Participants expressed a variety of other reasons why they *would* want their recordings deleted (Table 4.2). Some felt that *"there is no reason to keep the recording"* because *"it has no*

[10]References to <*Company*>, <*Device*>, or <*Assistant*> were automatically populated based on the participant's device.

[11]For example, *"there was no information worth keeping"* (P64), *"it doesn't need to be saved"* (P74).

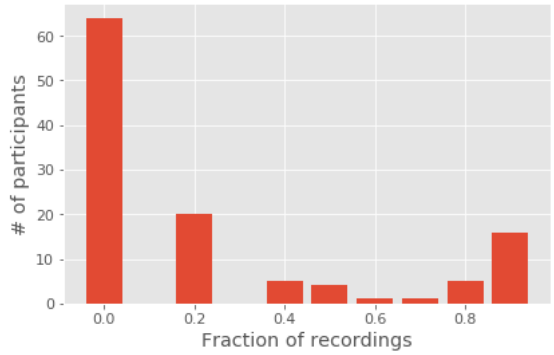


Figure 4.4: Fraction of recordings each participant wanted to delete.

value to me" (P78), while others wanted to protect certain data or information: *"I don't want it storing my Spotify information"* (P5). Accidental recordings were also a frequent deletion candidate: *"This was not even supposed to be heard by Alexa"* (P46). Not all respondents needed a specific reason for deleting an interaction; some simply felt uneasy about their data being stored for extended periods of time: *"I simply do not want this recording out there. It has been a while since I have used this functionality so it has been out there for a long time"* (P110).

Some participants decided whether they would delete a recording based on its perceived sensitivity. For example, P18 chose to keep a recording because *"it is just a basic request and conveys no personal information or interest (other than my voice pattern I suppose). So I would [not] feel the need to, and the system seems to work better when it has more recordings to help it learn/recognize my vocal patterns."* However, they stated that they would delete another interaction *"since this is info about my interests and preferences,"* acknowledging, though, that *"this type of info about me is already available in many different ways."*

In general, however, such heterogeneity appeared relatively rarely among our participants, and most adopted an all-or-nothing approach (Figure 4.4). A slight majority—55.2%—did not want to delete any of the recordings they were presented with. A sizable minority—13.8%—wanted all of their recordings deleted, regardless of their content. One participant summarized the attitude of those who fell into the latter camp: *"though this particular recording doesn't include any private information, I would like them to delete all of my recordings soon after they make them"* (P43).

Current privacy concerns

Our participants' reasons for deleting their recordings also shed light on what people consider sensitive. This can be gathered, for example, from the 5.8% of respondents who wanted to delete recordings because they considered them private. In some cases, partic-

ipants simply stated, *"This was a private conversation"* (P79), while in others they specified more about why they considered it off limits: *"no need to know what someone in my house like of music"* (P46).

An equal number of respondents (5.8%) expressed concern that information in the recording might help the voice assistant company build up a detailed profile of them, which they considered undesirable: *"I simply do not like to expose my preferences to things and have them analyzed. I would fear that such recordings could come up as Ads and create more unnecessary clutter in my internet experience"* (P16).

Stronger and more common than either of these themes was another privacy concern: children. Of those who chose to delete a recording, 13.5% mentioned that a child's voice was captured as their reason for doing so. Though the contents may be similarly innocuous to what an adult could have asked, the fact that the speaker was a child put the recording in a different category for some: *"I guess I feel differently about this one because it's a recording of my child's voice"* (P38). At least one respondent was distressed to realize that their child's voice was being stored in the cloud: *"I am having a reaction to having my granddaughter's voice stored somewhere"* (P67). Participants were likewise protective of recordings that included guests, with 7.69% choosing to delete a recording for this reason: *"It's a very common command that smart speaker users issue but since it was of a guest, then I may eventually delete it"* (P28).

To dig more into people's privacy concerns, we asked all our participants: **In the past, have you had any privacy concerns about your device?** Most participants (71.7%) said they had had no concerns about their smart speaker. As with many privacy-focused surveys, a common refrain was *"I am not a person that really ever has privacy concerns. I have nothing to hide and nothing worth stealing"* (P31).

Among the 28.3% of participants who said they had experienced privacy concerns, these were frequently caused by accidental activations: *"There were times when the speaker would activate without me saying the wake word. This was a bit odd and it did leave me a bit uneasy"* (P28).

Another common source of unease was the idea that the device might always be listening: *"just the ambient listening about what we talk about scares me. I wonder what data the device is collecting and what they plan to do with it"* (P89). One respondent implicated the government: *"I did wonder if it was just constantly listening and recording everything into an unknown database for government agencies. Probably does"* (P39).

While a number of participants expressed their trust in Amazon and Google—*"I trust Google a fair amount and have filled my home with these Google devices"* (P96)—others feared that the corporations' profit motives makes them poor stewards of privacy: *"I am not convinced that either Google or Amazon are committed to privacy if surveillance has a profitable up-*

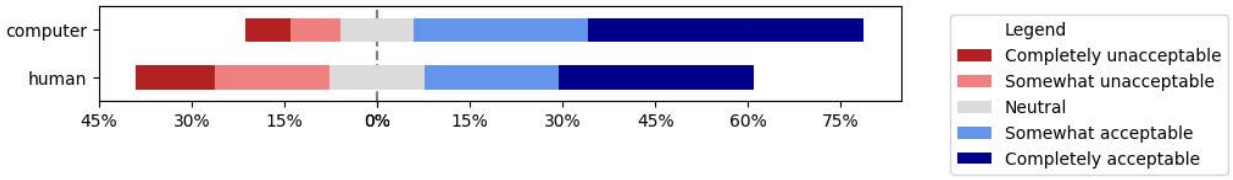


Figure 4.5: How acceptable would it be for this audio recording to be processed by a computer vs a human?

side" (P90). More concretely, two participants expressed concern that recordings of them may have been used for personalized advertising: *"There are some occasions where we will be talking about things without using Alexa and they will come up in Amazon recommendations or ads shortly after"* (P33). Finally, one participant wrote that what the companies actually do with the recordings remains opaque to them: *"While I guessed no one at Amazon listened live, I still don't know if anyone reviews them or how that data is secured"* (P43).

Feelings about different sharing scenarios

Our results so far have shown that the majority of participants are not particularly concerned about the privacy of their prior interactions with their smart speakers. However, this does not mean that people are apathetic about their privacy or that any usage of their existing data would be considered appropriate. Instead, people's acceptance of the status quo is tied closely to what is happening (or what they believe is happening) with their data [136]. This is revealed by the answers to questions where we posed some alternate scenarios and use cases about how data from the voice assistants might be used.

Who performs quality control? A commonly stated reason for why voice assistant companies retain people's recordings is that they need them to ensure quality and train better models. As seen above, users are aware of this use case and often support it, since they would like their assistant to work better. But how is this done and who gets to see the data in the process?

According to journalists' investigations, workers employed by Amazon are "tasked with transcribing users' commands, comparing the recordings to Alexa's automated transcript, say, or annotating the interaction between user and machine" [61]. However, privacy policies do not clearly state that other humans may be reviewing users' recordings, and, when we ran our survey, this fact remained secret from the public. Furthermore, our prior research about microphone-enabled Smart TVs has shown that a large fraction of users believe that humans will *not* have access to voice recordings from their devices [139].

To gauge the acceptability of these practices, we asked our participants: **How acceptable would it be for this audio recording to be processed and analyzed by:**

- **A computer program performing quality control for <*Company*>?**
- **A human, working for <*Company*>, performing quality control?**

While most respondents (72.8%) found processing by a computer to be acceptable, there were twice as many recordings (31.3% versus 15.3%) where respondents considered it unacceptable for a human to review them (Figure 4.5). Fisher's exact test (computed using a binary coding[12] of each participant's average acceptability score) showed that this difference is statistically significant ($p = 0.00960$).

Other Use Cases Improving the assistant's functionality is just one possible use for the interaction data. To gauge users' reactions to other potential use cases, we asked them: **How would you feel if <*Company*> used this audio recording for...**

- **Improving the assistant's performance, functions, or services**[13]
- **Providing you with additional functionality powered by <*Company*>**
- **Providing you with promotional offers from <*Company*>**
- **Providing you with additional functionality powered by other companies**
- **Providing you with promotional offers from other companies**

The results (Figure 4.6) showed that there were significant differences between how people viewed each of the scenarios (Cochran's Q, $p < 0.01$). While improving performance and developing new features was usually deemed acceptable (74.0% and 66.1% of recordings, respectively), using the audio for promotional offers (i.e., advertising) was considered unacceptable for nearly half of recordings (48.7%), especially if the ads were from third-party companies rather than the manufacturer (64.6%). Approximately half also negatively viewed the possibility that their recordings may be used to power functionality offered by third-parties (49.7%).

To see if people have different preferences for the usage of transcripts of their interactions, compared with the audio recordings, we also asked: **How would you feel if <*Company*> used only the transcript (not the recording) of this interaction for**... the same purposes as in the previous question. The results (cf. Figures 4.6 and 4.7) were largely identical.

[12]For binary coding of Likert scales, we split participants into those who found the usage "somewhat" or "completely unacceptable" (coded as 1) and everyone else (all other answer choices coded as 0).

[13]Since what exactly constitutes improved services is inherently ambiguous, we asked, in a separate question, **if <*Company*> said that they were using your recordings to "improve the device's performance, functions, or services," what do you think that would mean?** Most respondents suggested use cases like analytics, better models, and improved understanding of different voices. However, notably, four respondents expected this language to be a code for advertising.

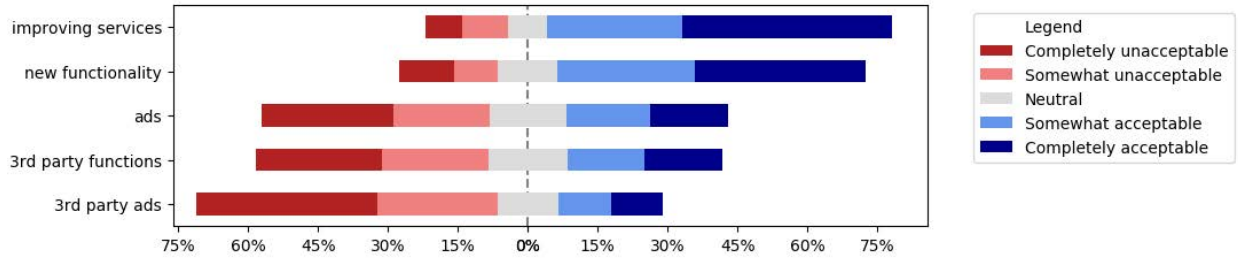


Figure 4.6: How would you feel if <*Company*> used this audio recording for…

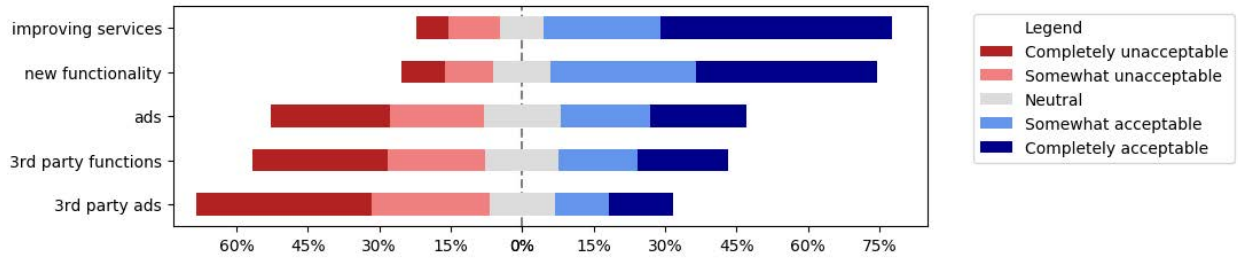


Figure 4.7: How would you feel if <*Company*> used only the transcript of this interaction for…

Existing privacy controls

Existing Alexa and Google Home devices include some privacy controls [10, 91]. We sought to understand how people try to protect their privacy with regard to their devices and whether they take advantage of the offered controls.

We asked participants an open-ended question about whether they had done anything to protect their privacy from the smart speaker: **In the past, did you take any steps to protect your privacy when using your device?** Only 18.6% of respondents described taking any steps to limit their devices. Among them, most commonly (43% of respondents who took privacy actions, 7.8% of all participants), users described turning off the microphone—*"Sometimes I would turn off the microphone especially if I was having a private, personal conversation with someone"* (P28)—or unplugging the device altogether: *"I've unplugged the damn thing when I know I'm going to be be having sensitive conversations in my home"* (P90).

Participants described modifying the devices' settings, for example to *"enable a[n] audible "beep" sound whenever Google starts listening"* (P7) or limit who can use the device: *"we only let people we know drop in on us"* (P25). Concerned about their children's privacy, one participant described *"making sure all of the do not save options are checked for my children's*

devices" (P81). Two participants said they *"do not let Echo make any purchases"* (P43). Finally, one person chose their device's location based on privacy concerns: *"I thought about putting one in my bedroom, but moved it"* (P14).

Familiarity with the review feature

One of the main ways smart speaker users can control their privacy is by reviewing their interactions through Amazon and Google's interfaces and deleting any interactions they do not want kept. However, the review interfaces are not necessarily well publicized, so people may not know about them. To find out, we asked our participants: **Did you know that there is a page on <*Company*>'s website where you can see the recordings and transcripts of all your past interactions with your device?** Our results showed that a majority of our respondents were not familiar with the review feature: 56.0% did not know it existed, compared with 44.0% who did.

The user experience for finding the review feature is different between Amazon and Google, and it is possible that the two companies advertise it differently. While a slightly higher fraction of Google users were familiar with the review feature, Fisher's exact test showed no significant difference ($p = 0.42$) between the two groups' familiarity with the review feature.

Knowledge of recording deletion

The review interface allows users to delete their interactions, but do they know this? We asked, **Were you aware that the review interface allows you to delete specific recordings of you interacting with your device?** Of the respondents who were familiar with the review feature, almost half (45.0%) did not know that they could use it to delete interactions.

Use of the review feature

To learn more about how people use the review feature, we asked those who had used it before: **Which statement best describes your use of the review feature?** Two thirds of participants who knew about the review feature reported using it on individual occasions, with an additional 5.9% stating that they do so regularly (Table 4.3).

To understand how this feature is used, we collected open-ended responses to the question **What usually prompts you to review your interactions?** Most reported examining their history out of *"absolute curiosity"* (P37). Another common reason for reviewing interactions was *"getting an inappropriate response, thus wanting to see if it's hearing me correctly"* (P26). For example, *"Alexa doesn't understand me and I want to see what she understood instead"* (P45). Several also reported that they *"went into the function accidentally"* (P15) while performing another task. One participant also used the history feature to recall informa-

I've reviewed my interactions on individual occasions.	66.7%
I know how to review my interactions, but have never done it.	17.6%
I regularly review my interactions.	5.9%
I know I can review my interactions, but don't know how to do it.	5.9%
Other	3.9%

Table 4.3: Responses to **Which statement best describes your use of the review feature?** as fraction of respondents who knew about review feature (51 people total).

I knew you could delete recordings, but have never done this.	67.9%
I've deleted recordings on individual occasions.	28.6%
I regularly delete recordings.	3.5%

Table 4.4: Responses to **Which statement best describes your use of the deletion feature?** as fraction of respondents who knew about deletion feature (28 people total).

tion from the past: *"when I am trying to remember the name of a song that I have asked her to play in the past"* (P98).

Similarly to the review feature, we wanted to understand whether and how people use the deletion feature. We therefore asked: **Which statement best describes your use of the deletion feature?** Two thirds of participants who knew about the deletion feature reported never having deleted recordings (Table 4.4). Only one person stated that they regularly delete recordings.

Since most people did not take advantage of the deletion feature, we asked those who were familiar with it but did not use it: **How did you decide not to delete any interactions?** Respondents universally agreed that *"there was nothing I felt the need to delete"* (P42) because *"our uses of Alexa are extremely mundane"* (P38).

To gain qualitative insights into the behavior of the 8% of our respondents who *have* deleted recordings, we asked: **What usually prompts you to delete interactions?** Participants generally reported deleting interactions *"if it's something private or just to clean things up"* (P114). Some also deleted interactions they thought could be considered embarrassing, like when *"occasionally it picks up something weird like me cursing at it"* (P18) or when a friend *"asked it some suggestive things as a joke"* (P9). P9 elaborated on their think-

ing: *"I did go and delete those things after the fact, but I really don't feel like it was necessary to have done so."*

The review feature in multi-user households

While the review feature can be an effective tool for controlling users' privacy, it may actually introduce privacy problems of its own. In multi-user households, where different people may be interacting with the smart speaker, the review feature may expose household members' interactions to each other—or, at least, the people who control the device. Depending on their awareness, people may therefore be inadvertently sharing search queries, listening preferences, reminders, and other personal data with others. This is especially concerning in light of recent reporting showing that the use of smart home technology in domestic abuse cases is on the rise [87, 33].

Some manufacturers provide tools to manage situations like these and enforce boundaries. For example, Google allows the initial user to add others to their device. It then differentiates between who is talking to the device using a feature called "Voice Match," which stores each person's interaction histories under their own accounts. However, activity initiated by unrecognized voices, including any guests, is stored in the "default" account—the first one to set up Voice Match.

To probe some of these inter-personal dynamics, we asked a series of questions, starting with: **When you were previously reviewing your device's activity, did you encounter interactions that were initiated by someone other than yourself?** Of those who reviewed interactions, 56.8% said they had encountered recordings of others while doing so.

We further asked: **Have you ever discussed with another home occupant or visitor a recording of them that you listened to?** Only 4 respondents (20% of those who had previously encountered other people's recordings) reported discussing interactions, with 3 of those cases involving children (P54: *"told my kids not to say certain things to Alexa"*).

Participants in our survey were typically the administrators of their devices, since being able to control the device was an eligibility requirement. But could others review their interactions? We asked, **Do you believe anyone else in your household has access to the recordings made by your device?** Most respondents, 71.6%, believed that no one else could access their recordings. However, many of these people mentioned that others in their household also have the Alexa or Google Home app installed when answering a different question, **Who in your household has the Amazon Alexa / Google Home app installed on their mobile device and/or linked to the main Amazon/Google account?** Depending on how the smart speaker is configured, having the app installed and linked to the device may be sufficient for obtaining access to the entire interaction history. Thus, up to 27.6% of our respondents may be confused about who has access to their recordings.

Feelings towards review by others

To understand how acceptable it would be to participants if others reviewed their interactions, we asked an open-ended question ($IRR = 0.918$): **How would you feel about other members of your household reviewing your interactions with device?**

Over three quarters of respondents (76.7%) said they had no concerns: *"I wouldn't mind at all"* (P34). Of these, about a third explained that they considered their interactions not sensitive: *"It wouldn't bother me. They would just get bored to death"* (P113). Among these, 11.7% further explained that they would not mind because they are a *"pretty trusting family"* (P87) or *"I have nothing to hide from my partner"* (P72). Some people (7.8%) were on the fence: *"Wouldn't really be a big deal, but would still feel rather odd"* (P26).

However, a sizable minority (25.2%) stated that they would be uncomfortable with others reviewing their interactions, writing that they *"would be quite perturbed by this"* (P90), calling it *"a sort of invasion of privacy"* (P44), and explaining *"I am not a freak but my interactions with Alexa is not something I would not like anyone else but me to see"* (P47). One participant gave a specific example of the kind of interaction they would not want others to review: *"I would be shocked because there are things that I do and look for that I absolutely do not want my kids to see or hear. Ordering their Christmas presents for example"* (P81). Given that we only recruited users who control these devices, we believe that the levels of concern we document are lower bounds: we would expect to see more concern among household users who do not control the devices.

Towards better privacy defaults

There may be opportunities for intelligent voice assistants to provide privacy defaults and controls beyond what is available today. We therefore surveyed our participants about a few alternative solutions.

Acceptable retention policies

Expecting users to manually review and delete interactions, which already number in the thousands, places an undue burden on them and is almost certain to result in most interactions going unreviewed. Furthermore, as we saw in Section 4.5, a significant fraction of users find permanent retention of their recordings unacceptable. A natural solution, also proposed in other privacy domains [110, 21, 204], is for the content to be automatically deleted after a certain period of time.

To gauge users' interest in such a policy, we asked: **Suppose the assistant had the option to automatically delete your recordings after a certain amount of time (that you specify). Do you believe you would enable this feature?** A large majority (77.8%) said they would be likely to enable this feature, and another 12.5% were neutral.

Open-ended feedback to this proposal was also overwhelmingly positive: *"sounds like a brilliant idea"* (P64). Even those who felt their recordings were not particularly sensitive were interested in this feature: *"It would be a good idea to clean this up to hedge against unintended consequences later"* (P115).

Since voice assistant companies may be unwilling to voluntarily limit their data collection, other parties may step in, for example third-party developers offering browser extensions to automatically enforce user-defined retention policies. To understand whether users would be amenable to these, we asked: **Suppose a third-party tool were available, which would automatically delete your recordings from <*Company*> after a certain amount of time. Would you install this tool?**

Over half (52.6%) said they would be likely or very likely to install such a tool; another 22.4% stated that they were neither likely nor unlikely to install it. Expressing a common sentiment, one respondent wrote, *"If Google does not implement the deletion of this data, and I can choose to have a browser extension installed, I would install the extension"* (P89). Participants were mindful, however, that installing such an extension could itself constitute a privacy risk and said they would take this into account: *"That would be fabulous. As long as the extension was secure and could be trusted"* (P67).

Issues of trust were on the mind of the quarter of participants who reported that they were unlikely to install the extension: *"A third-party tool or browser extension isn't guaranteed to be secure and my privacy could be endangered"* (P65).

So how long should companies store users' data? While there is no one correct answer, we surveyed our participants to look for trends and a possible consensus: **In your opinion, how long should <*Company*> store your data before deleting it?**

A quarter of respondents (25.8%) preferred the current retention policy, with no fixed deletion schedule and companies storing data until it was manually deleted. A further 9.5% of participants felt that the decision is best left to the manufacturer and were fine with their recordings being stored "as long as the company wants to." The remaining participants, nearly two thirds, wanted their recordings deleted after a fixed period of time. The retention period desired by respondents ranged from one hour (7 participants, 10.7%) to two years (2 respondents), and the median was 28 days.

Content detection and filtering

As we have seen, not all interactions with smart speakers are considered sensitive, but some are. One hypothetical privacy control would be to automatically delete recordings in categories users consider sensitive. With advances in natural language processing, concerted research could make this a realistic proposition. We therefore asked our participants: **Suppose the assistant had a feature that let you automatically screen out certain**

Personal/sensitive queries	20.4%
Nothing	16.5%
Children	15.5%
Financial topics	11.7%
Background noise	10.6%
Personal identifiers	9.7%
Queries on sexual topics	9.7%
References to locations	8.7%
Everything	7.8%
Other specific personal information	7.8%
Medical subjects	6.8%
Searches	6.8%
Specific people speaking	5.8%
Specific people mentioned	5.8%
Queries that reveal your schedule	5.8%
Everything during certain times of day	5.8%
Guests	5.8%
Embarrassing content	4.9%

Table 4.5: Fraction of respondents that said they want this type of recording automatically screened out ($IRR = 0.694$).

recordings and prevent them from being saved. Which characteristics would you want it to screen on? (Table 4.5).

Types of recordings many wanted screened out included any recordings of children (15.5%) – *"I would want my children screened out"* (P21); financial information (11.7%) – *"Any commands around shopping and banking including items, account numbers, passwords, bank names, etc."* (P40); accidentally captured conversations (10.7%) – *"I would want it to delete anything that is not directly speaking to Alexa for a command. Any extra conversation should be deleted"* (P77); queries on sexual (9.7%) or medical (6.8%) topics – *"topics of a personal or sexual nature"* (P39); as well as locations (8.7%) or other personally identifying information (9.7%) – *"Anything regarding my home address, travel destinations, or sensitive personal information, I would want to delete"* (P83). Participants were generally positive about this hypothetical feature; 39.8% of those surveyed stated that they were likely or very likely to use it, with another 35.0% remaining neutral.

4.6 Discussion

Our study's results shed light on people's beliefs, attitudes, and privacy preferences for smart speakers.

Ignorance of privacy controls

Nearly half of the respondents in our survey did not know that recordings and transcripts of their interactions are stored forever, and the majority did not know that they could review their interactions. Many were surprised by both of these facts. Almost no one (less than 3% of all participants) regularly reviewed their interactions. Even among those who were familiar with the review feature, almost half were not aware that they could delete recordings from the interface.

In addition to the review interface, smart speakers also allow users to disable their microphone by pressing a physical button on the device. However, in our survey, only 5% of participants mentioned using this feature, with another 4% describing how they simply unplug their device. As a result, we conclude that existing privacy controls are underutilized. Future research should investigate why this is the case and whether other controls would be more useful.

Lack of awareness appears to be one major reason. For example, responding to our survey question, **in the future, do you intend to take any steps to protect your privacy when using your device?** almost a quarter of respondents (23.8%) wrote that they intend to take actions based on information they learned from our survey: *"Honestly thank you for this survey. I would have never knew about the recordings and transcripts. I feel like I need to be more cautious"* (P99). This may be rectified, in part, by greater education (by the companies themselves or the media).

Disagreeable retention policies

When examining concrete interactions they had had with their devices and considering how long voice assistants should be storing them, most participants chose a retention period far shorter than the current default, which many described as unacceptable.

Instead, most respondents stated a preference that voice assistants adopt shorter retention periods. Almost 80% of surveyed participants said that they were likely or very likely to enable this feature if it were offered. Thus, we believe voice assistants would better align with their users' preferences if they deleted their recordings after a certain period of time. Researchers studying other kinds of personal data have made similar recommendations for retrospective data management [110, 21]. Such a policy would also align with the storage limitation and data minimization principles of the European General Data Protection Regulation (GDPR).

In fact, after our study concluded, Google announced new controls that allow users to opt in to automatic data deletion after either 3 or 18 months [150]. (Media reports also suggest that Assistant "quietly changed its defaults to not record what it hears after the prompt 'Hey, Google' " [80].) We consider this a step in the right direction but observe that nearly half of participants (49.1%) chose a retention period shorter than three months. Furthermore, users' low awareness of the review interface, and the fact that people in general strongly adhere to defaults, means that measures like this are unlikely to make a meaningful impact unless they are on by default rather than opt-in.

Amazon also introduced new privacy tools after our study ended, enabling users to delete a day's interactions with a voice command ("Alexa, delete everything I said today") [94]. However, users must navigate the app's settings to enable this feature [66], and recordings not deleted in this manner remain stored forever.

The Role of Privacy-Enhancing Technologies If voice assistants choose not to implement automatic deletion, our survey suggests that there is room for privacy-enhancing technologies to step in: over half of respondents stated that they were likely to install a third-party tool to regularly delete their interactions. However, respondents clearly recognized the trust implications of providing third parties access to their data, so any such tool would have to come from a trusted party and face auditing of its security and privacy properties. While self-reported data and the hypothetical nature of the offered service may mean that respondents overestimated their likelihood of adopting any technology, we believe our data nonetheless shows a strong demand for more privacy options.

Nuanced privacy perceptions

While users expressed a clear preference for shorter retention periods, they did not feel that their currently stored recordings presented a grave privacy danger. The overwhelming majority did not consider their interactions sensitive, describing them as "mundane" and stating anyone perusing them might get "bored to death."

However, respondents felt markedly different about stored interactions of people other than themselves. Participants were particularly protective of the privacy of their children, choosing to delete recordings that included children's voices and stating their desire for recordings with kids to be automatically filtered out (*"they are too young to understand or consent to that,"* explained P18). Participants also wanted recordings of guests to be removed (*"I would want recordings of any guests to my house deleted,"* P73).

In general, then, people seem more protective of the privacy of others (*"I'm sure my roommate doesn't want this recording just floating around,"* P110). Future work can test this hypothesis more directly. If evidence is found, this can be the basis for novel approaches to privacy interventions: asking people to choose privacy policies and settings for others, rather than themselves.

Unacceptable secondary data uses

Despite not considering many of their recordings sensitive, participants were very clear that the use of their data for advertising purposes would be unacceptable.

Respondents were also largely uncomfortable with third parties gaining access to their data, even for benign purposes. Data sharing for the purpose of "providing you with additional functionality powered by other companies" received as much disapproval as the advertising use case, even if only the transcripts of interactions are shared. This is particularly noteworthy as both Amazon and Google allow developers to integrate with their voice assistants, and these third-party "skills" can be invoked without the user naming them directly [58].

Multiple users create tensions

Many smart speakers are installed in households and environments where they are used by multiple people. The review feature offered by voice assistants therefore introduces a new privacy risk: household members may learn about each other's queries and interactions, which would have otherwise remained private. Most respondents in our survey were not overly concerned about this happening, reasoning that *"they're around to hear most of them anyway"* (P83). Furthermore, as we have seen in our results and prior research [24], controlling media and other relatively mundane requests represent the majority of present-day usage. Still, a quarter of participants shared that they would be uncomfortable if others had access to their interaction history. However, our results suggest that, for up to 36.2% of participants, others might in fact have such access.

While today voice assistants provide some controls for multi-user environments, our survey found scant evidence that these are being used: no respondents mentioned Amazon Households, the program that allows users to add additional accounts (including special ones for children) to their device, and only one participant referenced Voice Match, Google's system for distinguishing speakers. Thus, we believe more effort is needed to design and implement effective privacy controls that would satisfy the needs of households where multiple people interact with a smart speaker.

Contextual integrity

Our results are consistent with the Contextual Integrity model of privacy, which posits that established social norms govern information flow in distinct contexts [160, 159]. An information flow consists of the data subject, the type of information being shared, how that information is being shared (the transmission principle), the sender, and the recipient (including their role and purpose of the sharing), and occurs within a specific context that is governed by norms and expectations. (See Chapter 3 for a more in-depth explanation.)

In this study, we found that smart speaker users are generally comfortable with the "default" context of voice assistant usage: queries being transmitted to Amazon or Google for the purpose of answering them. However, any deviations from these defaults immediately cause concern among many people, for example, if there are changes to the subject (recordings of children or guests rather than the device owner), purpose (advertising instead of answering queries), or recipient (third parties instead of the manufacturer).

Other factors, such as how long recordings are stored, constitute the transmission principle. Our results show that many people find this transmission principle important, considering certain storage policies unacceptable. Future work should examine additional transmission principles in greater depth. As smart speakers evolve, system designers should use the lens of contextual integrity to examine whether potential changes would create inappropriate information flows and, consequently, be considered privacy violations.

Lessons for passive listening

While this study focused on people's privacy attitudes about existing smart speakers, many of its results may be applicable to passive listening devices as well. Our survey demonstrated that there is demand for more effective filtering of recordings, in order to screen out accidentally captured conversations and topics considered sensitive. Similar filtering will be even more important for assistants with passive listening. Other examples of features that can benefit both types of devices include more effectively distinguishing speakers (including unfamiliar ones and especially children) and identifying conversation topics (and eliding irrelevant and possibly sensitive remarks).

Another lesson for passive listening is the desire expressed by many in our study for more limited data retention. While there are good reasons to retain data—it allows users to review past interactions with the assistant, especially if something went wrong—its accumulation presents a privacy risk and makes users uncomfortable. This will be even more true with passive listening devices. System designers will need to think more deeply about the tradeoffs inherent in collecting and keeping this data and must provide more options to users for deleting it automatically and in a timely manner.

Chapter 5

Laying the Groundwork for an Always-Listening Permissions Architecture

This chapter examines the design space for passive listening permissions, elaborating on the assumptions, establishing a threat model, and discussing the approaches available for achieving privacy. Its key contributions are a detailed methodology for studying passive listening devices and a taxonomy of privacy protection approaches.

5.1 Introduction

In Chapter 4, we layed the foundation for passive listening permissions by studying people's attitudes about their present devices. Next, we wanted to commence studying passive listening assistants themselves. However, as we have already discussed, these do not exist. We need a much more concrete idea of what we are controlling, before we can start designing the controls themselves.

We began explaining our assumptions in §1.3 by discussing the general capabilities of a passive listening assistant as we envision it. In this chapter, we elaborate on them, discussing in more detail the features we expect a passive listening assistant to have and how they compare with existing ones. With this clarified, we define the threat model that, informed by the notion of contextual integrity (Chapter 3), will guide our design.

After that, we present an overview of the privacy protection strategies that we can pursue. Since there are many choices, characterizing them here allows us to begin comparing them for effectiveness. We will continue the comparison in future sections (Chapter 6) and choose several of them to study in greater depth (Chapters 7 and 8).

5.2 Scope & assumptions

Trying to impose limits on the behaviors of an always-listening device inevitably raises questions of trust. Suppose we develop a "permission system" for always-listening devices. Where should it run? Is it external to the smart speaker? If so, who will operate it? Why are they to be trusted? How will it stay up-to-date? How will it work when the voice assistant takes other form factors? Alternately, the privacy controls may be built in to the device itself. But can the platforms really be trusted to police themselves?

These problems are vast and complex; to make progress in this space, we begin by defining a threat model where the smart speakers themselves are trusted, as are the platforms that operate them. These platforms, however, are open to third-party applications, and these are the ones from whom we want to protect users' privacy.

In other words, under our threat model, the manufacturers and their software are fully trusted with all audio their devices may overhear. This does not mean these platforms are incapable of violating users' privacy; indeed, there are a multitude of ways for them to do this, either for their own purposes (e.g., selling information about users) or for others (for example, state-sanctioned backdoors or surveillance). Yet, in trying to define a trusted computing base, the line must be drawn somewhere, and the hardware manufacturer is a reasonable place to start. We hope that eventually these assumptions can be relaxed, but for now, we will trust the hardware and its operating system, including any first-party functionality, and assume that they can safely hear everything that happens around the device.

Whom *don't* we trust, then? We believe that, as they are today, assistant platforms will be open to third-party developers, who will create apps that provide certain functionality which requires an always-listening smart speaker. If such an app were a standalone device, it would receive all audio from the microphone—24/7—and if it happened to be malicious, it would use that audio towards some nefarious ends. Our goal is to develop a system that can prevent this sort of abuse by enforcing limits on the audio an always-listening app may access.

We believe this is a realistic and practical threat model. First, it matches how existing ecosystems have developed, for example, mobile operating systems rely on central app stores to offer third-party-provided functionality. This also matches how existing voice assistants have positioned themselves: both Amazon and Google already allow third-party "skills" for their voice assistant and encourage their creation by, for example, offering hundreds of thousands of dollars in prizes to developers [79].

Overall, this architecture is likely to be more beneficial to privacy, as there is a single trusted party controlling the microphone. The alternative is for each "smart" device to have its own microphone and custom logic about when it listens and how it reacts. This

means that each device would be responsible for privacy on its own. As we know from the sad state of IoT security [199, 174, 127], this may not be ideal.

Some further simplifying assumptions that we will make—in the interest of making the scope of the problem more manageable—is assuming that the passive-listening devices in question will be targeted at consumers (rather than businesses), located in a home environment, and fixed (rather than portable). However, we note that passive listening in the workplace presents its own interesting set of challenges, from both a technical and security perspective, some of which have begun to be explored in the literature [142].

5.3 Current voice assistants

Smart speakers with passive listening apps are unlikely to just show up, in the near future, in industry showrooms and on store shelves. The technology, and specifically the current state of natural language processing, is simply not ready yet, though it is advancing rapidly. Consequently, we may reasonably expect a relatively gradual progression in functionality from today's intelligent voice assistants to those with more passive capabilities. As such, it may be useful to review the current behavior of voice assistants and smart speakers, as a sort of baseline for any future developments.

Large numbers of people have embraced smart speakers and other intelligent voice assistants: reports suggest that 86 million smart speakers were sold in 2018 alone [219]. Intelligent voice assistants come embedded in a variety of devices: smart speakers and displays (e.g., Amazon's Echo and Echo Show), laptops, smartphones, smart plugs, and many other consumer electronics [222, 36]. Regardless of the specific form factor, these share two things in common: an always-on microphone and an Internet connection.

An on-device speech model has been pre-trained to identify the assistant's wake-word among the ambient sounds. When the wake-word has been recognized, the device records the subsequent audio until a pause in speech or a timeout has been reached. The speech is then sent to the assistant's web servers, where it is processed, analyzed, and the requested action (if understood) is triggered.

The exact processing pipeline used by popular IVAs remains proprietary, but common steps include automatic transcription (going from speech to text), domain detection (understanding the general type of query; e.g., it is about travel), intent detection (recognizing the user's specific goal; e.g., booking tickets), and slot filling (inferring the parameters of the query; e.g., destination and date for the tickets) [83]. These steps may not necessarily happen sequentially; feedback from later stages (such as slot filling) may result in updates, for example, to the understanding of the intent. Alternately, the stages may happen all at once, with a single model trained to perform domain detection as well as slot filling [175].

In addition to one-shot interactions (a request or query followed by a response), IVAs have also started introducing more complex interactions. Some may keep a small amount of state [183]: for example, after inquiring about an artist, a user can ask "when were they born?" without naming the person again. IVAs can also engage in rudimentary dialogue, usually asking for confirmation or follow-up questions. Notably from a privacy perspective, when a follow-up question is asked, the microphone "opens" (i.e., starts recording) directly, without hearing a wake-word.

The audio, its inferred transcript, and the assistant's response are stored by the companies indefinitely by default. Users are able to review these interactions and delete them, though many do not know about these capabilities (see Chapter 4). Besides this, the only privacy control available on a device is a physical mute button or switch. This is typically available on smart speakers, but fewer other assistant-enabled devices.

Third parties and their capabilities The major voice assistants today allow third-party developers to provide additional functionality for the voice assistants. These are called "skills" for Alexa and "Actions" for Google. To implement this functionality, developers are provided with a declarative API, which they use to list and provide examples of the intents their app fulfills and the slots (parameters) they expect to fill. Most often, users must invoke a skill directly, e.g., "Alexa, tell SmartHome App to turn on the lights." However, certain skills may be invoked automatically, without the user uttering their name, if the platform detects a user's intent as matching the one implemented by the app [58]. When an app is invoked, it is provided with the parsed data as well as a transcript of the original utterance. At present, third party apps for both Amazon and Google do not get access to the underlying audio for their requests.

Privacy and security challenges Existing smart speakers pose privacy problems along a variety of dimensions. The device must be trusted to only listen for its wake-word (and record only after it is said) and not the rest of the time. Even if it does this faithfully, accidental activations can occur, often without the user's knowledge; in some cases, this has even led to entire conversations being shared with third parties [189]. Adopting our threat model (i.e., focusing only on the risks of third-party apps), possible attacks include "skill squatting" [118] (malicious apps impersonating legitimate ones by adopting similarly-sounding names). Skills where the assistant automatically selects the appropriate app based on the request ("ok Google, hail me a cab") may also be targeted by attackers for impersonation. Such attacks, as well as the information targeted by attackers and their motivation, may carry over to always-listening apps as well.

5.4 App capabilities and designs

Our goal is to envision privacy solutions for passive listening apps, but none exist today which could be analyzed for their properties and capabilities. Therefore, as a first step, we

will define what these capabilities may be. Understanding the variety of designs and use cases for these apps will help us ensure that any proposed solutions are able to address the full spectrum of features these apps may develop.

What functionality will apps provide?

Since we are discussing technology that is largely hypothetical, we can only speculate on the needs developers may seek to address. We have come up with examples that—while far from exhaustive—are plausible and representative of the potential use cases. To generate these examples, we consulted literature on how people currently use smart speakers [24] and perused existing third-party skills for assistants [9]. Additionally, in a study I conducted with other researchers, we surveyed 178 people about their expectations for services that always-listening assistants would provide [200]. To ground participants' suggestions in real-life scenarios, we had them read conversations excerpted from various linguistics corpora, which recorded people talking as they went about their daily lives. We then asked participants about potential services an always-listening assistant would provide. (We also studied how they would feel sharing the details of these conversations with the assistant.)

The list of apps that resulted from this process can be found in Appendix B, along with utterances or conversations that may trigger them. Some examples include:

- A calendaring app, which picks up on plans you make and adds them to your calendar automatically
- A foreign language learning tool, which can help when you're speaking a foreign language by correcting mistakes or suggesting vocabulary
- A "swear jar" to keep track of how many times you say undesirable words
- A kitchen helper, which can keep track of ingredients and answer questions about recipes
- "Artificial memory" that saves and organizes information (such as names and birthdays) mentioned in a conversation for easier recall
- A baby monitor can alert you to when your child is crying (or, when they grow older, if they're fighting with a sibling)
- A music DJ, which adjusts the music in the room based on the mood of the party (as inferred from the conversations)

What Audio Do Apps Need to Provide?

The examples above represent a range of use cases for always-listening apps. What audio would each of them want to capture to provide the necessary features? This depends on

an app's specific functionality, of course, but more generally on:

1. How are apps invoked?
2. How do apps interact with users?

How are apps invoked?

Techniques can range from today's direct invocations via wake-words to fully-passive listening.

Direct address with wake-word This is the way assistants and their skills are triggered today; for example, "Mycroft, turn on the lights." While we expect this to remain the most popular way of invoking apps, existing paradigms may offer sufficient privacy protections. In particular, the fact that users are directly naming the app they want to summon means privacy violations are likely to happen only in the event of accidental invocations.

Flexible trigger-word invocation A variant of the wake-word approach is allowing users to construct more natural-sounding queries where the trigger word need not be the first word in the sentence: "can you turn on the lights, please, Alexa?" While the addressee is still semantically clear, identifying it requires more sophisticated analysis.

Call to action without trigger word A variant of the familiar app invocation may happen when a user issues a call to action that is not preceded by a trigger word. For example, instead of saying, "Computer, louder" or "Computer, pause" the user may want to simply say "louder" or "pause."

Purely passive (just listening) Apps may listen for speech that is not directed at them, but may still be relevant to their functionality. (We refer to these as "passive-listening apps," in contrast to the broader category of always-listening apps, which may use any of the other invocation modes.) For example, they could keep track of the conversation to have context if they are eventually invoked (e.g., knowing which song people are discussing when they decide to request information about the artist), or they could take actions silently (e.g., making a note that the household is out of milk based on someone's comment, without speaking up and interrupting the conversation). Compared to other invocation modes, the relevant speech here may be longer, have fewer keywords, more ambiguous grammatical structures, and require more context.

How do apps interact with users?

Another important variable to consider is the kind of feedback apps provide to users: do they respond immediately or take an action in the background? This design decision has implications for a user's ability to detect an accidental or privacy-violating invocation.

Provide immediate responses This is how today's voice assistants behave. This provides an opportunity for immediate feedback (audio or visual) that a particular skill was invoked.

Engage in dialog with the user In addition to providing an immediate response, the app may ask for confirmation or clarifying questions. This has the privacy advantage of providing immediate feedback, but is technically challenging, as the app must maintain state and context.

From a privacy perspective, the implication of this interaction mode is that the system has to not only detect the initial invocation, but understand whether any follow-up utterances are directed at the app. (This is another classification problem that malicious apps could try to exploit.)

Background action or no response In this mode, the app does nothing, or performs an action through a channel that is not the system itself. Conversely from the previous mode, this might have challenges as far as feedback is concerned, especially in combination with the passive listening (non-)invocation mode.

How are new apps installed?

Other considerations in the design space include whether apps are installed through a visual interface or by talking to the device. The latter case presents additional challenges, as it significantly limits the channels for presenting information to the user, such as a privacy notice or any additional information about the app.

5.5 Attacker model

So far we have discussed the behavior and capabilities of benign apps. However, some apps may be privacy-invasive, whether accidentally or due to malevolence. Our goal is to protect against these malicious applications. To do this most effectively, it will help to understand what constitutes an attack, the attackers' motivations, and the information they may target.

Defining attackers and privacy

As the goal of our system is to ensure users' privacy, an attack is any action that results in a privacy violation, and an attacker is someone who engages in these actions. Of course, this merely invites the next question: what constitutes a privacy violation?

Traditional permission systems, such as those used by smartphone operating systems, focus privacy controls on a handful of "sensitive" data types, such as a user's location

or their contacts. However, in reality, there are many more types of information whose leakage may be considered a privacy violation by users. This is because it is impossible to divide information into "sensitive" and "not sensitive." People freely share even "sensitive" information, like health facts, in certain situations, e.g., with doctors, support groups, and families. On the other hand, they may consider their purchases not sensitive, yet object to the sharing of their shopping habits. These apparent contradictions make sense if we observe that people feel upset when their information is shared in a way that runs counter to their expectations.

This notion is formalized in Nissenbaum's theory of privacy as contextual integrity (CI) [160], see Chapter 3 for more background. The contextual integrity model has important implications for our quest to design privacy controls for always-listening assistants. It suggests that it is not enough for our system to prevent access to the data types enumerated above (even if this were feasible). Instead, it must strive to ensure the contextual integrity of conversations: that the applications only get access to the data that is relevant to their purpose, and everything else remains off-limits.

Contextual integrity also illuminates a type of privacy violation that may be almost impossible to prevent: information collected for one purpose being used for another. A simple example is data collected for the purpose of fulfilling functionality being repurposed for advertising. This violates contextual integrity because the information flows beyond the original recipient to a third party for an unintended purpose. If the data in question is speech and the recipient is an always-listening app, our platform may be powerless (from a technical point of view) to curb such further spread of information once it has left the system. Yet, this is still a very important consideration, since any such privacy violations will necessarily impact the overall trustworthiness of the system as a whole. This problem is likely most amenable to non-technical solutions: careful vetting of developers and terms, contracts, and legislation that specify consequences for misuse of data.

Attacker motivations

Contextual integrity suggests an intimidatingly large set of privacy violations: anything outside established norms of information flow. However, an attacker generally does not set out to violate privacy for the sake of violating privacy. Instead, they are guided by their own personal (or organizational) reasons [125]. If we understand these motivations, then we may be able to deploy more targeted defenses. Thus, it is worth considering the question: what are attackers trying to achieve?

Marketing In today's economy, a common cause for privacy violations is the developer's desire to collect user data for advertising purposes [176]. The data may include specific facts (e.g., location, gender) or inferences that can be made about the individual (e.g., interests, income) [145]. This information may be used directly by the developer or sold to a data broker. Marketers are already seeking to obtain this information from

voice channels [207], so we expect similar motivations will drive many privacy-violating always-listening apps.

Theft of private information Another common motivation for malware is theft of secrets that can be valuable on the illegal market [73]. This is typically financial information (e.g., credit card numbers, bank accounts) but can also include personal identification numbers (e.g., Social Security Number or equivalent) and account usernames and passwords. An always-listening device is likely to eventually pick up this sort of information, making it a particularly attractive target for attackers.

Profiling household occupants It is possible that software may want to profile and understand the habits of the people living in a household for purposes other than advertising. Example scenarios include a utility or government agency wanting to ascertain how many people are living in a particular household or a criminal syndicate deciding which houses to burglarize. While the latter appears far-fetched (it does not scale, and the economics appear questionable), the former is drawn from real-world privacy concerns surrounding smart meters [30].

Reputational damage Some attackers may be motivated by causing reputational damage to their victim—whether a specific individual (e.g., politician, celebrity) or a class of people based on some characteristic or behavior [125]. For this type of attacker, there is no single "data type" that they may be trying to collect (though any data type from above may be considered useful); instead, lifestyle or ambient information may be targeted.

Unintentional privacy violations Finally, it is important to consider that privacy violators may actually be well-intentioned, and their violations are entirely accidental (or, less charitably, due to negligence), for example, due to a poorly-trained classifier. In fact, this is currently the primary source of privacy violations by voice assistants [186, 65], and we expect mistakes to remain the dominant cause of privacy violations due to inherent difficulties in natural language processing (and software engineering more generally). How do we prevent these mistakes from happening? Defending against more targeted attacks can help protect against accidental ones as well; however, there may be more specialized techniques that can be used if we assume an app is simply confused rather than malicious.

Information targeted by attackers

Having examined attackers' motivations, we now consider whether there is certain information they may be especially driven to obtain.

Indiscriminate data collection One way an attacker may seek to satisfy their goals is by collecting any and all data they can get their hands on, storing it, then mining it for useful information later. (This may include any speech, including non-primary languages, as well as nonverbal sounds.) The difficulty of accomplishing this will depend in large part

on the design of our always-listening platform.

Targeted collection Indiscriminate collection is harder to conceal, especially in the presence of any counter-measures, and may also be difficult to scale. As such, attackers may limit their collection to only data they consider valuable (based on their specific motivation). The following attempts to enumerate the specific data types attackers might be after:

- Demographic details, such as location, language, age, gender, income, etc. (to enable targeted advertising)
- Financial and other secrets (bank details, credit card numbers, passwords)
- Security measures (physical—do you lock your doors?—and digital: what operating system you use, whether you use two-factor authentication, etc.)
- Brands you use (for market research)
- Personal details, such as names, dates, pets, phone numbers, etc. (for social engineering)
- Health events (for advertising, as well as insurance purposes)
- People in the house (see discussion in §5.5 above)
- Timing of in-home events (to correlate with other data; less plausibly, for burglaries)
- Political opinions (for advertising, or social control)
- Controversial behaviors (for social leverage)
- Criminal acts (confessing to a crime)
- Crying or other signs of abuse (to share with government)[1]

Attack types

The final question we consider is how the attacker will try to collect the information we outlined above. In the case of accidental "attackers," they will collect data through the same methods, but not on purpose. We envision several methods of attack. (Note that each strategy makes certain—possibly contradictory—assumptions about how the always-listening platform operates and may therefore not be applicable to all architectures.)

[1]Note that this type of data sharing may be considered a privacy violation even if it is in the public interest. System designers will therefore need to navigate the attendant ethical questions when deciding how to handle these scenarios.

Direct listening The attacker may choose to directly listen for the personal (or other) information they are interested in, without any concealment, obfuscation, or other trickery. In doing so, their hope is that the review process does not catch their behavior.

Alternately, attackers may pursue a strategy that provides them plausible deniability and lowers the chance of detection. In general, these approaches can be thought of as *overly broad listening*.

Capturing things when the classifier has lower confidence Generally, an NLP model will have some confidence score indicating to what extent a user utterance is likely to match a particular intent. One simple strategy is to capture audio even for lower-than-expected confidence scores, in the hope that something useful (for the attacker) is captured.

Always find named entities relevant If the app performs its own named-entity recognition, it could always deem the entities relevant to the app's purpose (either because they legitimately do not know or maliciously).

Relevant keywords in irrelevant contexts For example, a flight-booking app hears you talking about drinking a flight of beers. As with other scenarios, benign false positives can be expected to dominate actually malicious behavior.

Homophones (similar-sounding words) For example, a malicious app might listen for "past word" instead of "password," thus evading a filter. This behavior lies at the core of "skill squatting" attacks on existing voice assistants [118].

Listening "around" appropriate times (past the end) A passive listening app may be privy to entire conversations, which it then attempts to process to fulfill its functionality. But what is the boundary of a topic or conversation? This is a hard technical challenge even for benign apps. Malicious ones may try use this as an "excuse" to keep listening even after the relevant conversation is over.

Inference / side channels An app could attempt, for example, to infer income by products used, travel patterns by timing, and so forth. Voice alone has been shown to be sufficient for inferring a broad range of characteristics, including age, gender, personality, physical conditions, and emotions [117].

5.6 Evaluation

Now that we have defined attacker goals and motivations, we can start thinking about how to design protections against them. But once we have a candidate system, how should we evaluate it? In this section, we consider how we might evaluate a potential platform against our goals of ensuring user privacy and security.

Rigorous evaluation criteria are needed because different approaches to assuring privacy will necessary involve trade-offs, and we need ways of comparing them. We propose that there are two broad axes for evaluation:

1. **Effectiveness**: how well does the system achieve its goal of increasing users' privacy?
   Metrics include: functionality loss, privacy gain.
2. **Usability**: how well would the system work for real people?
   Metrics include: usability comfort, trust, acceptability, surprise, and reviewer effort (if applicable).

Effectiveness seeks to formalize the notion of how well a system achieves its goal of preserving users' privacy. Since this is the primary purpose of the system, it is a core metric for success and evaluation. Of course, there may be multiple ways of measuring effectiveness—and different components to it.

However, the system's *usability* is also crucial. The notion of usability captures the extra burden a privacy-enhancing system places on users, in terms of time, effort, cognitive load, and other expenses. Usability can be at tension with effectiveness, as a system that zealously guards a user's privacy may bother them with warnings, notifications, and needlessly blocked false positives. On the other hand, usability can be seen as contributing to the system's overall effectiveness: an unusable system will not be welcomed by consumers or adopted by companies. Usability too can be broken down into further subcomponents.

Evaluation metric details

The metrics that measure the effectiveness of the platform are functionality loss and privacy gain.

Functionality loss

This metric can be formulated as: "if app A is denied access to resource R, it will lack functionality F." Given clear assumptions about apps, it should be possible to evaluate this metric in an automated manner (see §5.6 below).

Privacy gain

While a bit more ambiguous, working with some definitions, again this metric may be evaluated automatically. However, there are a few different ways of measuring it:

Absolute privacy gain Without the platform, any app would have access to 24 out of 24 hours every day. Suppose that the platform limits some app to listening for only one hour in a given day. (The rest of the audio is blocked from reaching the app because it is considered irrelevant.) This translates to 23 hours of "absolute" privacy gained. However, this figure does not account for *which* audio the app retained access to—it could be that the most sensitive information was not successfully shielded.

Privacy gain for sensitive conversations Consider an app that, without the platform, only used 2 hours (out of 24) of data. The platform detected that one out of those hours was sensitive and made it off-limits, resulting in a one-hour gain.

Privacy gain based on user inputs An app provides functionality for cars, flights, and hotels, but the user wants to only use it for flights.

For both functionality loss and privacy gain, we expect to encounter false positives and false negatives, because language is ambiguous and permissions are necessarily coarse. False positives mean the platform allows an app access to some speech, even though the app does not need and may not even want it. False negatives mean there is speech that an app needs for its (proper) functionality, but is denied access to by the platform.

The functionality loss and privacy gain metrics, as defined above, can be empirically measured in an experiment based on evaluating simulated passive listening apps on the proposed platform design. The first step, then, is to select sample apps. While it may not be necessary to implement their functionality in full, the experiment requires classifiers that are *equivalent* to those of the apps, in that they should correctly classify speech (or text) that the apps will consider relevant and use for their functionality.

The next step is to generate or collect labeled speech examples. These should include both negative examples (i.e., speech not relevant to the app; this should be relatively easy to source as most things are likely to fall in that category) and positive examples (which may need to be generated specifically for the app). If the platform assumes that the speech will be transcribed before reaching the app, then the examples may be text-only.

Next, run the app's classifier on the sample speech. Achieving high accuracy is important because our goal is to use this as a proxy for what a real app *needs* access to for functionality; anything the classifier returns *true* on will be considered functionally necessary.

The final step is to "apply" the platform or permission system to the sample apps and rerun the classifier, now limited by the platform, on the sample speech, measuring what it still has access to. The functionality loss and privacy gains can then be inferred from these results.

Usability

Usability is traditionally measured by how well users are able to perform specific tasks. This includes how easy it is for users to find the information they want and choose between apps (i.e., make an informed choice about which app to install, based on the permissions it requests). Below, we survey several of the components that go into making a system usable.

Time and effort How much time is required to make a decision and install an app? How much effort (including cognitive load) is involved in the process?

Comfort, trust, and acceptability How willing are people to install apps with this system? For example, users may be uncomfortable with a totally opaque solution that outsourced screening to the app store. Chapter 6 focuses on evaluating this metric.

Reviewer effort (if applicable) Certain platform architectures may rely on reviewers (professionals or community members) to examine apps for compliance with standards and adherence to rules or its own declared behaviors. In this case, an important metric for the success of the platform is the amount of time (as well as money and effort) that each review is expected to expend.

Surprise

If there is a mismatch between the audio an app gets and a user's expectations for what it should hear, we refer to this as "surprise," and it should be the goal of any architecture to minimize this. There are several ways in which surprise may occur.

Caused by bad or confused ML For example, imagine that a user says, "I got into a fight," but the system hears "flight" and captures it. This would be surprising but perhaps not interesting from a system-security perspective (unless this was a deliberate attack, along the lines of those discussed above in §5.5). We therefore generally consider this problem outside the scope of our threat model, assuming perfect recognition and machine learning as much as possible, while acknowledging that reality falls far short of these assumptions.

Caused by ambiguously specified or overly broad permissions For example, a user gives permission to an app for booking flights, but is surprised that the system captures their conversation about birds flying. In situations like these, both the developer and platform could be at fault: the platform for having overly broad categories, or the developer for choosing too broadly. It could also be a sign that a malicious application was successfully able to evade the platform's privacy protections.

To measure the level of surprise induced by a proposed privacy platform, the following experiment can be used. Show people triples of {app, permission, example utterance}.

That is, provide a description of the app as it would appear on the platform, as well as any information about the "permission" it requested or resources it would have access to. Sample speech, as in the effectiveness experiment, should be drawn from a corpus of positive and negative examples. For each sample app, ask whether they expect the app to receive that utterance given the permission. If they say no, the app has failed at communicating what is in scope. In other words, it is surprising, which is bad! It is important that this evaluation includes positive examples that the app will not get due to the permissions (if these exist); these are the most interesting data points. Additionally, participants can be asked to flag speech they think the app will get, but should not be allowed to.

5.7 Architecture approaches

At this point, we have considered passive listening apps and what they may look like, how and why attackers may seek to exploit them, and what an evaluation might look like. But we have remained, as much as possible, agnostic to the details and mechanisms a system will use to protect users' privacy in this new paradigm of passive listening apps. Now, we begin to examine the architectural choices a privacy-protective platform may rely on.

Content-based vs. metadata-based controls

Today on mobile platforms and personal computers, privacy controls and permission systems operate primarily based on metadata, such as the source of the data, the sender, or the recipient. For example, firewalls can block access to certain ports or entire devices, ad blockers prevent certain domains from receiving data, and file permissions restrict access to users with sufficient privileges. In all of these cases, the system is agnostic to what the underlying data is; it makes its determination based on facts it knows *about* it. Even smartphone permission systems, which ostensibly distinguish between data types like contacts and location, actually operate by restricting access to specific resources and APIs, rather than inspecting the data flows. More general techniques, like access control lists, capability-based controls, and information flow control are also based on metadata.

However, when we are dealing with passively listening applications, and the data is speech, metadata-based systems are insufficient. Two conversations between the same two people may have identical metadata—yet the content will be different, and therefore the conversation could be appropriate for one app but not another. To enforce user preferences for always-listening devices, privacy controls must be able to make this distinction. They therefore have no choice but to go beyond the metadata. Thus, furnishing users with this type of choice requires a solution that draws on a different type of approach: allowing (or disallowing) access based on the content of the communication.

Applying this paradigm in practice will require either advances in speech recognition, if controls are applied on-device, or a much greater degree of trust in platforms, if controls are applied in the cloud. On the bright side, once this paradigm is sufficiently advanced, these methods will be applicable to other domains (e.g., replacing the microphone permission on smartphones with more granular controls).

Inspecting a transmission and using its contents to decide how to treat it is not unprecedented, and there may be lessons to learn from prior instantiations of this paradigm. Deep Packet Inspection does exactly this, with a variety of Intrusion Detection Systems relying on this technique to detect attacks. Anti-spam techniques also rely on examining the message body and, in some cases, even use natural language processing to detect unwanted messages.

The multi-level security (MLS) system used by the US military for handling classified information is another interesting example of content-based controls. Under this scheme, documents (as well as their individual sections) are assigned labels based on the sensitivity of the information they contain (confidential, secret, or top secret); an individual's clearance level determines whether they are allowed to access the document (their clearance must be at least as high as the document's classification). Since a document's classification level depends on its content, the system as a whole represents a type of content-based controls. A vast body of research, including a variety of formal methods such as the Bell-LaPadula Model, has examined how to track these labels through a system [29, 62]. However, at that stage the problem becomes another type of metadata-based control, as the system is content-agnostic once a label has been assigned. The content itself is only relevant during the actual classification process. This is a manual (and labor-intensive) operation that entails following a classification guide and checking at what level each piece of information that needs to be communicated should be classified [209]. A number of classification guides have been made public [210], but they are understandably focused on securing government secrets rather than everyday communication. Still, there may be lessons to learn from this system; for example, it may be possible to divide personal information into tiers based on its sensitivity, with apps only being granted "clearance" to certain tiers.

What are content-based controls? The basic principle of content-based privacy controls for always-listening devices is that they will examine some amount of speech, and decide whether to deny an application access to it or allow it through (possibly after certain editing or transformations). At the most general level, there are two families of approaches a system to protect privacy may take. We refer to these as **deny-listing** and **allow-listing**.

Deny-listing This approach relies on the observation that many people do not consider most day-to-day speech sensitive, and attackers are likely to be after only certain data types (see §5.5 above). Therefore, it may be sufficient for the platform to identify speech that is truly sensitive and prevent apps from getting access to it.

Allow-listing Taking the opposite tack, this approach envisions that apps' access is *scoped to capabilities*. In other words, apps should only get access to what they need to function. We conceptualize this as a permission system: similar to permission systems in modern smartphone operating systems, apps must declare ahead of time the resources they wish to use, and the platform enforces these restrictions. This approach achieves greater privacy benefits, since it would limit applications to hearing only the speech they need. Furthermore, as discussed in §5.5, the contextual integrity model suggests that this approach better aligns with user expectations. Therefore, in the remainder of this chapter, we will consider how we might build such a system. (In practice, however, we expect systems to use a combination of deny-listing and allow-listing techniques. Allow-listing can guide the behavior of most apps most of the time, and deny-listing can act as a safe-guard to prevent the leakage of data considered categorically sensitive.)

What do permissions look like?

In considering the design of a permissions system for a passive listening app, we must address several related questions:

How do apps declare permissions? In smartphone permission systems, app developers specify upfront which permissions their app requires by including a list of these permissions in the app's manifest. This is a straightforward approach that works well; adopting it for always-listening apps could be one way forward. However, for this to work, we must be able to explicitly enumerate the resources a developer may require. This presents challenges with respect to natural language understanding: for example, how exactly do we determine if something is considered health-related? An additional challenge comes from the way natural language processing models currently work: by relying heavily on neural approaches and training on a wide range of examples. In light of a neural network's probabilistic nature, developers who have trained on a variety of conversation examples may struggle to pick out a specific list of permissions that is associated with them. Therefore, we must consider alternate approaches. For example, a developer could be required to submit their app—or just its natural language component—for testing and evaluation.

After solving the challenges of how apps declare their requirements, a permission system needs to address the questions of user involvement: how—if at all—is this information conveyed to the user, which choices do users have, and how much customizability are they allotted? Each of these questions has crucial implications for security and usability.

Do users get a choice? The overarching consensus in the field of usable security is that the burden of making security decisions should be removed from users as much as possible, replaced instead with security (and privacy) by default [56]. Applied to the problem of always-listening apps, this maxim would suggest that an ideal system would ensure that all apps behave in a privacy-respectful manner. Yet, even if this were possible, there

would still be an element of user choice: deciding whether to use an app or not. Any privacy-sensitive user would make this decision in part based on what the app is likely to hear, and this will be seeking information to answer that question. They could obtain it from the developer's description of the app, however this may be vague or incomplete. This is where the permission system can come in: by providing a clear and standardized way for a potential user to understand an app and its privacy impact.

How are permissions presented to users? If a permission system decides to share information about the apps with its users, the next question is: how? As pointed out above, permissions declared by apps might not be human-readable. Even if they are, there might be a gap between apps' true behavior and how concepts are interpreted by users. Therefore, care must be taken in ensuring the understandability and usability of the interfaces presented to people.

When are permission requests presented to users? A particular issue in sharing an app's permissions with users is when to present that information. One natural point is at the time of installation. However, research in smartphone permission systems showed that users rarely paid attention when presented with install-time permissions. This motivated the move to "ask-on-first-use" in mobile operating systems [74, 90]. That system would present its own challenges for always-listening voice assistants, as it may be unclear when an always-listening app is first "used." Furthermore, with audio as the only output channel, the ways an app's permissions could be presented to users are further restricted.

Should users be able to deny or limit permissions? Denying means not allowing certain permissions at all (e.g., an app asks for 5 permissions, but the user only grants 4 of them). Limiting means restricting permissions within an existing hierarchy; for example, if a skill wants access to "travel" but a user wants it to know only about flights. This feature is desirable, but it may add significant complexity to the platform, both from a technical and cognitive perspective, as it would necessitate a user interface for reviewing permissions and for users to make active, informed decisions.

What is the role of other parties in the system, such as app store reviewers or superusers? What choices, capabilities, and responsibilities might we assign to them?

In the next section, we consider and compare several approaches permission designers may take. Each will vary along the dimensions above.

5.8 Permission system design space

There are many ways to design a permission system. Rather than exhaustively listing all potential variants, here we want to sketch out the different *approaches* one might take.

In comparing and understanding the approaches, it may help to set out our *design goals*. While they may not be as quantifiable as the general evaluation metrics defined in §5.6, they clarify the principles that guide our designs. These include:

Data minimization The goal of the allow-listing family of approaches is to select speech that is relevant to an app's functionality and appropriate to its purpose—and only that speech. The data it gets is thus minimized. By definition, then, data minimization is a core objective of a allow-listing-based permission system.

Transparency To gain users' trust, we believe it is imperative for our system not only to act correctly, but to be transparent with users about what is happening with their speech. Therefore, it is important for users to know, which speech of theirs an app would gain access to.

Consent Users should provide positive consent before an app is installed and thus obtains access to their speech. However, it remains an open question how that consent should be obtained, and what level of interaction and review this process should entail.

Leverage the wisdom of the crowd We believe that, while it should always be up to an individual to decide whether an app is right for them, we can leverage other people to help in this process. For example, dedicated workers, volunteers, or other users could help determine whether an app's behavior is appropriate. This information could then be shared with prospective users or automatically used as part of the evaluation process of an app.

With these in mind, several potential approaches follow. We note that each approach may work well for certain use cases and not at all for others. This is expected, and a real-world system may employ them in combination. Our goal here is to understand their advantages and disadvantages.

Approach: keywords

Looking for keywords in text: only allow sentences that include certain words. This is reminiscent of how today's voice assistants work [105], except the trigger word does not have to come first.

This approach is obviously insufficient for many purposes but may be the right choice for certain use cases, such as requiring the keywords "remind," "forget," or "remember" for a reminder app.

Challenges Natural language has many ways of expressing an idea, so creating an exhaustive list of keywords may be difficult for any non-trivial task. (And over-eager keyword lists may create privacy problems of their own.) Furthermore, keywords may have

homophones, homographs, and different meanings of the same word. In general, this approach is not applicable to many use cases, especially passive ones.

Approach: allow-listing topics

This involves pre-defining "buckets" that speech might fall into. For example, a conversation might have a topic [5], or a sentence might have an intent [132]. The system would allow apps to subscribe to certain buckets.

Challenges For developers to choose a topic for their app, the platform must maintain a reasonably exhaustive list of all conversation subjects human speech may plausibly contain. How can this list be generated? Is it even realistic? Even broad categories (e.g., "food") may generate privacy gains, which is good, but how should the system handle conversations—and apps—that cover multiple topics? Individual sentences can contain multiple, potentially conflicting, topics or partial matches. And many passive use cases cut across topics or are based on higher-level concepts.

Sub-approach: allow-list entities / slot filling

Rather than subscribing to topics, apps would declare the "types" of things they're looking for:

- A unit conversion app is looking for two unit measures plus a quantity.
- A travel app is looking for two destinations and a date.
- A mapping app is looking for a location.

Challenges Not all apps may be able to formulate their inputs as concrete entity types. Additionally, some inputs may be implicit in speech and never uttered out loud; for example, the origin or destination of travel might be implied by the speaker's current location. This limitation can perhaps be addressed if the app could provide on-demand feedback to the user about what it heard and which details were missing.

Approach: embeddings

Word embeddings are a type of NLP technique that maps words or phrases to a point in vector space [206]. While each dimension does not necessarily have semantic meaning, similar words or phrases end up clustered together. For the permission system, apps could declare the regions from which their speech should come.

Challenges This approach assumes that all skills would use the same intermediate representation. Recent advances in transfer learning in the NLP domain have made this more plausible, but it's still not clear how realistic this approach is. This approach also suffers

from low explainability. On the other hand, it may be combined with other approaches (like "transparency", §5.8 below) to counteract this, and it is notable in enforcing stronger guarantees than some of the other approaches.

Approach: derived features

Have the platform compute "features" over people's speech and expose these to apps, instead of the speech itself. Examples of features may include language, speaker, tone, or sentiment [131]. For certain use cases, these may provide sufficient information with few or no details about the underlying speech being necessary. This bears strong similarities to topic allow-listing (if topics can be considered features) and embeddings approaches.

Challenges What are the features that the platform should make available? Furthermore, most of the problems from the related approaches are applicable here as well.

Approach: network-restricted mode

This follows a slightly different paradigm from the previous approaches: *listen locally and tightly control exfiltration*. The app runs on the device or in the manufacturer's sandbox with full access to all speech. However, it can only talk to the outside world through limited, pre-declared interfaces. The permission system (what users review/allow/deny) is defined over these interfaces—i.e., the outgoing data—rather than over the underlying speech. For example, a calendar app is limited so that it only sends dates over the wire. These permissions could either be predetermined at install-time, or dynamically requested—and subject to user review—during the app's runtime. (We will explore the latter possibility, in much greater detail, in Chapter 7.) Network-restricted mode could also be used in conjunction with other permission approaches, for example by allowing users to turn it on only for certain periods of time or for certain apps.

Challenges A full-featured calendar app (to continue with this example) will want to create events that have, at the very least, a title and, ideally, a description with additional information. All of these are open-ended text fields which, by default, are not subject to control. This creates an opportunity for major leaks, though the platform can try to enforce stronger guarantees about the contents of such fields. However, for truly malicious apps, this still leaves open the possibility of leaks through side channels (e.g., exfiltrating data through the pattern of requests).

Potential mitigation One way to combat this sort of attack, even those that rely on side channels, is by resetting state. The concern with network-restricted mode is that you can exfiltrate arbitrary speech in open-ended fields (e.g., event title). Suppose, however, that the classifier/detector only got access to one sentence at a time. (It might be allowed to carry over state from previous sentences, but presumably be restricted in what that state

is.) Then, it would be limited to exfiltrating that single sentence as the event title, but not anything else from the larger conversation context.

Approach: transparency

In this approach (which we will study in greater detail in Chapter 8), apps submit a machine learning model that is a speech detector/classifier. Its job is to classify all speech into one of two categories: whether or not it is relevant to the app. When installing an app, users see examples of conversations (drawn from a corpus maintained by the platform) and whether the "relevance detector" classified them as relevant to the app. Users can also try their own examples.

Challenges This approach requires a static model, but developers may want to frequently retrain their models to improve performance. Asking users to constantly re-review their apps is not usable. Other concerns include edge cases, obscure examples (not covered by the platform's speech corpus), and models trained in an adversarial manner, specifically to avoid detection.

Sub-approach: crowd-sourced review

Developers supply a human-readable description of the kind of speech their classifier aims to capture. This is compared to the examples actually captured (as in the transparency approach). For example, Mechanical Turk workers may look at the examples and decide if they match the description or not. Then, users only need to review the app descriptions, rather than any speech examples. This also allows for more frequent updates to the relevance models.

Sub-approach: bootstrap from prior examples

Like transparency, but actively restrict apps from accessing speech that is not similar to examples that have been allow-listed or explicitly okayed by other users.

Challenges How is similarity measured? And who determines it? Answering these questions well determines whether this is a viable approach.

Sub-approach: runtime opt-in

When the app wants to capture speech, request the user's consent (similar to ask on first use permission models in smartphones [205]).

Obviously, this is not feasible for every request, but it could be used in combination with the bootstrapping approach, or when other permission approaches have low confidence.

Moving forward

The approaches outlined above represent a (no doubt incomplete) characterization of the solution space for designing a privacy-enhancing platform for always-listening devices. These solutions need to be studied, ideally under realistic use cases, to understand when they work and circumstances under which they struggle. The subsequent chapters begin this exploration, first by comparing the approaches against each other based on their acceptability to end-users (Chapter 6), and then with deep dives into the Network-Restricted (Chapter 7) and Transparency (Chapter 8) approaches.

Even before our investigations, intuition suggests that no approach will be totally sufficient on its own. Another question, therefore, is how to combine these approaches into a single system that takes advantage of their strengths rather than multiplying problems and security holes. These can be further combined with solutions that already exist or have been suggested for existing voice assistants, such as physical mute buttons, automatic deletion (see Chapter 4), and "sleep words" that temporarily disable a device's listening capabilities.

Furthermore, no solution is likely to be "one size fits all," because of varying privacy preferences and risk tolerances among users. Therefore, a device should be able to customize its operation to a particular user. Due to the changing nature of the technology, this process should be continuous: the device should always be learning about what the user considers acceptable and what they consider to be creepy.

Chapter 6

Understanding the Acceptability of Different Permissions Approaches

We had participants install always-listening apps from a simulated app store, under different privacy conditions, to gauge different approaches' acceptability and see whether that affected people's propensity to use a passively listening assistant. We found a preference for stronger privacy controls, which was reflected in app install choices. However, regardless of their preferences, most participants did not naturally pay attention to the apps' permissions.

6.1 Introduction

In the previous chapter, we discussed a variety of privacy permissions approaches (§5.8)—such as allowing speech based on keywords, limiting speech based on topics, and restricting an app's network communications—and several ways to evaluate them (§5.6) based on their effectiveness and usability. This chapter focuses on one specific evaluation: comparing the permissions approaches by studying the *acceptability* metric introduced in §5.6. Specifically, our research question is: **which privacy protection approaches make people more likely to use passive listening assistants?**

Experiment design

Our goal is to determine which privacy approaches would make people more comfortable with using passively listening voice assistants. One approach is to ask people directly: describe the assistant and the potential privacy protections, then survey self-reported preferences and willingness to adopt the technology. But, in addition to this, it may be more

instructive to obtain consumers' revealed preferences: seeing how they actually make decisions, rather than asking about how they *say* they would.

In general, there are a few ways in which preferences about voice assistants could be revealed. First and most obviously, people who are more comfortable with passive listening would be more likely to get one of these devices. They would also be more likely to install apps for it. (Recall from §5.2 and §5.4 our assumption that features in our assistant are provided by third-party apps.) Once the apps are installed, those more comfortable would be more willing to use them (i.e., keep the assistant enabled and have conversations around it).

Of course, here we once again run into difficulties because passive assistants are still hypothetical: this makes revealing preferences about them challenging. However, the first and especially second observations from above are still somewhat measurable. In other words, a fully working intelligent voice assistant does not necessarily need to exist for us to be able to ask people questions about whether they would be willing to get one or install apps for it. For this reason, we decided to focus this study on the app installation process for passive listening assistants.

Framing our study around app installation enhances the realism of the preferences we collect, since this process is more similar to how users will encounter information and make decisions in the real world: they will not be required to read multiple informational pages of a survey, but they *will* (more likely) need to install apps. Therefore, to make this study more realistic for our participants, we created an app store interface, modeled on existing smartphone app stores, where people could browse and install apps for their assistant. The interface was the same for everyone, but the privacy information it displayed varied depending on the privacy condition a participant was assigned to, enabling between-subjects comparisons.

When considering the app installation process, we hypothesized that **when people are more comfortable with an assistant's privacy approach, they will install more apps for it**. This supposition is based on the inverse of that statement: if people are less comfortable with a device's privacy features, they will be less likely to install new apps for it, because doing so might expose their private speech to untrusted third parties.

Based on that hypothesis, this study compares the number of apps installed in different privacy conditions. However, we caution that the link between app installation and comfort is unproven; and therefore we note that the null hypothesis (finding that there is no difference between number of apps installed in different privacy conditions) does not imply that people have no preferences between the offered privacy protections. (It could be that such preferences exist, but do not translate into different numbers of installed apps.)

Because of this uncertainty, and to enable us to verify the hypothesis above, we designed

our study to collect people's stated preferences about the privacy conditions in addition to their quantified behavior (the number of apps they installed). To facilitate this, we added a within-subjects experiment to our study, in which we had each participant try out more than one privacy condition. This allowed us to see how people thought about them, and we could ask directly to what extent the privacy differences influenced their decision.

Privacy models

Our next task was to determine which privacy approaches to include in our acceptability experiments. All of the 5+ approaches described in §5.8 would be suitable for inclusion in this study. However, because we were using novel methods with an uncertain effect size, we decided to limit the number of conditions in our study and focus our evaluation on just two approaches to permissions—the ones we considered most promising—as well as a third, control, condition.

Accordingly, we defined three privacy models that would serve as the independent variable in our study:

- The **Control** model has no privacy protections. Every app would have access to all audio heard by the voice assistant. (For example, a flight reservations app might hear and share *all* conversations happening in the home.)
- Under the **Topic** model, speech that falls into an app's category will be accessible to the app. A category is a predefined "bucket" of speech such as a topic or intent. (For example, the topic for a flight reservations app might be *travel*.)
- In the **Network** model, the voice assistant processes audio locally and will only pass speech on to third-party API endpoints when necessary for an app's functionality. (For example, a flight reservations app might need to share destinations and desired prices with its server, but all other speech in a conversation can stay local.)

(Table 6.1 lists expanded descriptions of each privacy model.)

6.2 Methods

To test our hypotheses, we designed a study that had participants browse a simulated app store under different privacy conditions and answer questions about their experience. In this section, we detail our methods for collecting and analyzing the data.

Study flow overview

We framed our study as a survey with an interactive component (browsing the app store). Our survey consisted of a mix of free response and multiple-choice questions. We used a

Table 6.1: **Privacy model descriptions**. the descriptions of the different privacy models, as seen by participants. "ALVA" is the name used to describe the passively listening voice assistant in our study.

	Description
Control	Every app has access to all the audio that ALVA's microphone picks up. This means that a conversation held in the same room as ALVA will likely be entirely heard by all apps, while a quiet conversation in another room behind a closed door probably won't be heard.
Topic	To minimize the audio shared with third-party apps, ALVA lists the topic of speech that the app has access to. Only speech that is relevant to that topic will be accessible to the app. However, all speech that falls into an app's topic will be accessible to an app, even if it is not explicitly relevant to the app's functionality.
Network	Apps and their third party developers are responsible for providing functionality to ALVA given a speech recording. For instance, a weather app might receive a location "Los Angeles" from ALVA and retrieve the weather for Los Angeles. ALVA will be listening to all speech that its microphone picks up, but will only record speech and send it to an app when necessary for functionality. This means that, even if ALVA heard you say something, nobody would find out unless it was sent to an app. What ALVA sends to each app is restricted by the app's privacy policy, which is defined by each app's third-party developer. This policy is found in each app's permissions section, which you can view before installing, on the app's ALVA Store page.

fictional new voice assistant, "ALVA," as a product that the participants would be giving their opinion on. A unique element of our survey is an interactive app store that participants could interact with. The purpose of the app store was to create an experience that would more closely replicate what users may experience in the real world. This may result in more organic data compared to, for instance, presenting all three privacy models together and asking for comments on them, which might lead participants to express opinions disconnected from actions they would take in a real situation.

At a high level, participants in our study went through several steps:

1. Learning about the concept of passively listening voice assistants
2. Learning about a specific privacy protection model
3. Installing apps from the interactive app store
4. Answering questions about their installation choices
5. Learning about a different privacy protection model, as offered by a "different version" of the device
6. Again installing apps from the interactive app store (for the new version of the device)
7. Answering questions about differences between the two privacy models and their choices

Thus, participants learned about two different privacy protection models. These were assigned randomly, without replacement (i.e., no participants experienced the same model twice).

Study details

We began the survey by collecting information about participants' prior usage of existing voice assistants (e.g., Siri, Amazon Alexa, or Google Assistant) and their present attitudes towards them.

Next, we introduced ALVA, explaining that ALVA is always-listening, which means that—unlike existing voice assistants—it does not need a wake word (e.g., "Alexa," "OK Google") to activate. We used an attention check question to ensure participants were reading the explanations; those who failed this attention check were excluded from further participation and were not included in the reported data.

We then introduced the idea of "apps," which add functionality to the otherwise featureless base voice assistant. Participants were randomly assigned to learn about one of the three privacy models (described in 6.1). At this point, we asked two more attention check

questions to ensure participants understood that they would need to install apps to use ALVA and how their model worked. Unlike the previous attention check question, we did not disqualify participants for failing these comprehension checks, but instead asked them to try again.

Participants were then given their task: they needed to browse the app store and install any apps that they thought they would use if they were gifted an ALVA device. The app store contained various apps such as alarms, music, and a calendar. In total, 18 apps were listed on the store homepage, along with their icon and name. (The complete list of apps in our study can be found in Table 6.2).

The store listed every app as having been created by a "third-party developer" (without specifying their name or other attributes). Additionally, all apps requested permissions appropriate to their functionality (i.e., there were no malicious or over-permissioned apps).

Participants could click on an app to view its page (Figure 6.1), which consisted of three sections: a general description of the app, a permissions section that described the privacy policy for each app, and an examples section with sample relevant phrases that would trigger a response from ALVA. Details in the "Permissions" section of each app's page in the app store corresponded to the privacy model in the condition the participants had been assigned.

Another element of the interface was the shopping cart, where people could view the apps they installed and uninstall them if they changed their minds. We recorded the list of apps installed, the pages visited, and the time spent on each page and section. We used this data to ask participants follow-up questions and, after the survey, analyze what participants had been paying attention to.

We did not enforce a minimum amount of time spent browsing the store, nor did we require participants to install any apps. (If they chose to install none, we prompted them if they were sure, to prevent accidental submissions.)

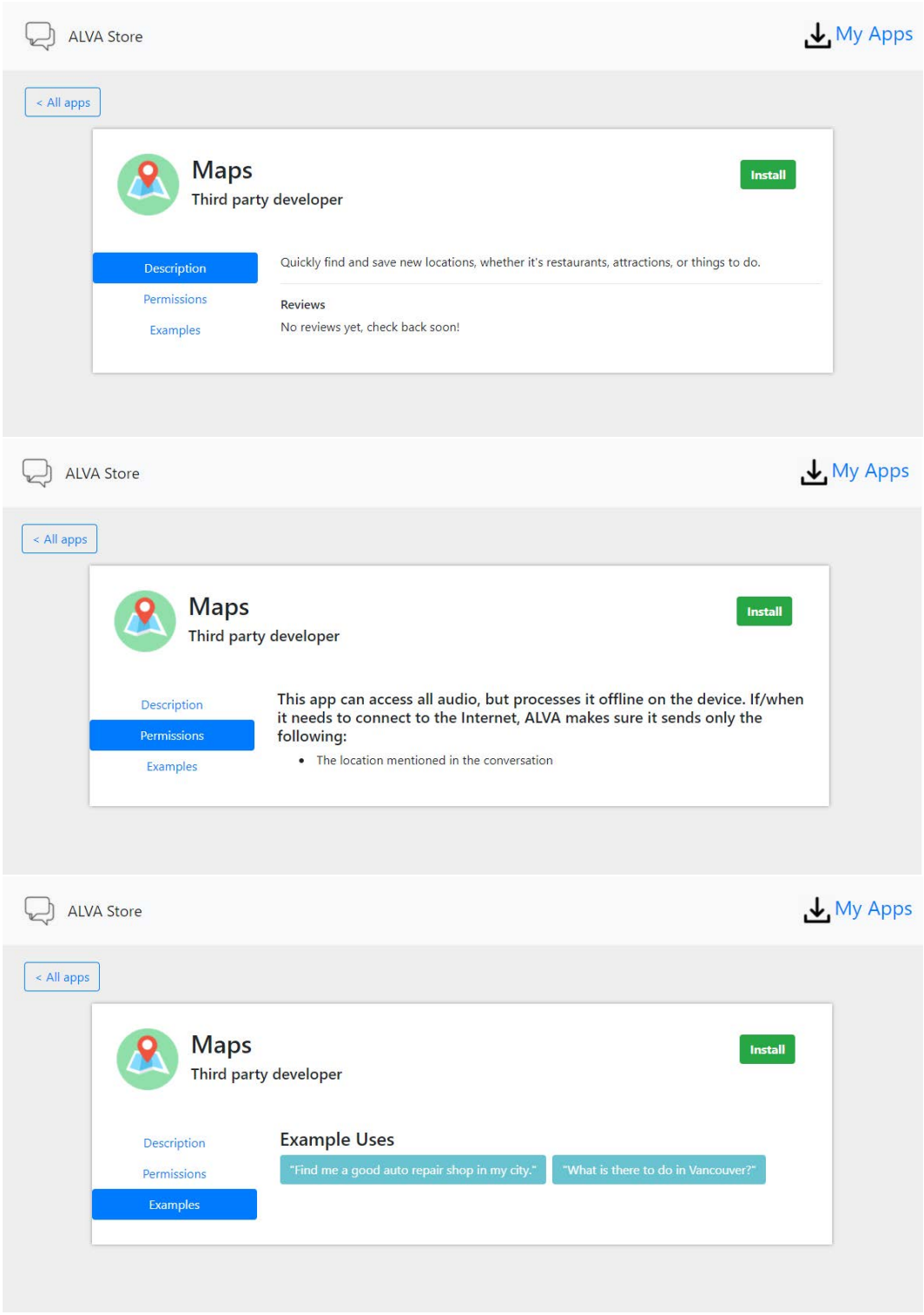
After participants indicated that they had installed all the apps they were interested in, we proceeded by asking some questions about the participants' experience browsing the app store. If they had not installed any apps, we asked why this was the case. Otherwise, we asked about their motivation for installing an app and for looking at an app but deciding not to install it.

Afterwards, we asked participants how likely they would be to use Alva, and why, how easy the found it to control information shared with ALVA and third party apps, and whether anything was unclear about ALVA's privacy model. For the likelihood question, we asked, "If you received an ALVA smart speaker as a gift, how likely would you be to set it up in your home and use it?." Participants could respond on a five-point scale

Table 6.2: **ALVA apps**: The apps our participants could install, along with their descriptions that were included in the app store.

Name	Description
Search	Queries you would ask Google/Siri/etc.
Calendar	Automatically add planned meetings and appointments to your calendar.
Reminders	Detects when you want to remember something and automatically save it, in order to remind you at an appropriate time.
Shopping List	Adds items to your shopping list based on your conversations.
Fridge Monitor	Keep track of items in the refrigerator and when they are likely to expire.
Recipe Search	Helps you quickly find recipes and dictates them to you, hands-free.
Baby Monitor	Detects and alerts you when your child is crying.
Alarm	Creates and sets custom alarms.
Coffee Scheduler	Prepares a custom cup of coffee in advance.
Weather	Get the weather forecast. This app can also sense when a conversation discusses plans that may be affected by the weather, and offer relevant advice.
Scorekeeper	This app keeps track of running total scores. For instance, this app can gamify household chores by keeping track of children's scores.
Swear Jar	This app keeps track of the number of times someone says an expletive. You also have the option of donating a certain amount to a charity of your choice after swearing too many times.
Trivia	This app lets you play a game of trivia against other people or ALVA.
Audiobooks	Play audiobooks out loud. Connect to third-party services to get access to all your digital books. Pick your voice, leave bookmarks, and easily switch to the part you want to hear, all hands-free.
Email	Check and send emails. Connect to your existing email accounts and stay productive, hands-free.
Maps	Quickly find and save new locations, whether it's restaurants, attractions, or things to do.
Meditation Instructor	Whether you need a few minutes to relax or want to practice daily mindfulness, a guided meditation instructor can help make this process smooth and simple. Choose your desired session length and what you want to focus on.
Music	Easily play music from your favorite artist and discover new music. Can tell you what the current song is.

Figure 6.1: **Screenshots of the app store interface**, including app page description, permissions, and examples sections



ranging from "Very unlikely" to "Very likely". For the effectiveness of control question, we asked participants to rate on a five-point scale ranging from "Strongly disagree" to "Strongly agree" to the following statement: "I think I can easily control the information I provide to ALVA and its third party apps."

After sharing their thoughts about this first iteration of the smart speaker, we then asked participants to imagine a different version of the assistant device, ALVA 2, that had one of the other two privacy models. They repeated the process of learning about the privacy model, browsing the app store, and giving their thoughts on the privacy model. This round, participants answered, if applicable, why they installed different apps in Round 1 and Round 2. They again described their likelihood of using ALVA, the perceived effectiveness of the privacy model, and the clarity of our explanations.

Hypotheses

We formulated several hypotheses about the effect that the availability of privacy controls would have on people's propensity to use the voice assistant.

1. The number of apps installed by participants will be greater in conditions with some sort of privacy protection.
2. Participants will show greater willingness to use ALVA if given a privacy model.
3. The effectiveness ratings of the privacy models will be higher than that of the control condition.

Analysis plan

Our analysis looked at the mean number of apps participants installed for each privacy model and participants' subjective preferences for which model they were most likely to use.

We used permutation tests to compare the difference in the mean number of apps installed between the three conditions. We chose this test over a Kruskal-Wallis/Mann-Whitney U as it better modeled the statistical assumptions of our study and did not require knowledge of the underlying distribution of the random variables. However, we also verified our results with these more traditional statistical tests.

The null hypothesis was that the number of apps installed was the same in each condition (i.e., had the same distribution). To test this, for each phase (Round 1 and Round 2) separately, we examined at which condition each participant was assigned to and how many apps they installed. We chose to analyze the phases separately as the data points are not independent, and the order in which the participants receive the conditions has an effect.

To perform the permutation test, we first calculated a base difference in sample means $d = \bar{X}_1 - \bar{X}_2$, with respective sample sizes n_1 and n_2, from this data for each pair of conditions (Control - Topic, Control - Network, Network - Topic). We then pooled the data into a single distribution. For 100,000 iterations, we permuted the data and computed the sample means for the first n_1 observations and the remaining n_2 observations. We then calculated the difference in these sample means, δ_i. The proportion of the permuted differences in sample means greater than our base difference $\delta_i > d$ is our p-value.

We compared how likely participants would be to use ALVA for each condition and how effective participants found each model by using a Wilcoxon signed-rank test for each possible pair of conditions. We took all participants who received the two conditions in any order and compared the likelihood and effectiveness ratings for each condition.

To analyze the responses participants gave to our open-ended questions, we used qualitative coding. We looked at all the responses, observed the major themes, and created a codebook of codes to categorize each response. Because a response could contain multiple sentiments, the codes were not mutually exclusive, i.e., a response could be labeled with multiple codes.

Participants

We recruited participants from the research recruitment platform Prolific to take an online survey. In total, we used data from 214 participants. The majority (57%) self-identified as male, the average age was 35, and the average household size was 2.8. Most (87%) of the participants had some prior experience with using voice assistants. The survey took approximately 15 minutes to complete, and participants were compensated $3.75 for their time.

6.3 Results

We hypothesized that people would install more apps in conditions in which they are more comfortable with the privacy model. To account for possible ordering effects, we separated the data into two phases. We refer to Round 1 as the first privacy model the participants were assigned to and Round 2 as the second. As shown in Table 6.3, the mean number of apps installed in Round 1 across conditions were essentially the same, with a standard deviation of $\sigma_{A1} = 0.123$. However, we begin to observe greater differences across conditions in Round 2, with a standard deviation of $\sigma_{A2} = 0.804$ apps installed.

To assess whether the difference in the number of apps installed in different conditions is significant, we used a permutation test (Table 6.4). Among all participants, the differences between conditions in Round 1 were not significant. In Round 2, we did find that participants installed significantly more apps in the Topic and Network conditions, as compared

Table 6.3: Mean number of apps installed by condition, all participants

	Control	Topic	Network
Round 1	5.5	5.8	5.7
Round 2	3.3	4.7	5.2

Table 6.4: Were there significant differences in the number of apps installed *by all participants* between each pair of conditions? Permutation test p-values.

	Control-Topic	Control-Network	Network-Topic
Round 1	0.47691	0.74139	0.72236
Round 2	0.01537	0.00100**	0.31023

Table 6.5: Mean number of apps installed by participants who viewed permissions ($n = 62$)

	Control	Topic	Network
Round 1	6.0	5.6	5.6
Round 2	2.5	3.8	5.7

with the Control. There were no significant differences between the Network and Topic conditions.

Next, we divided participants into subgroups based on their browsing behavior and investigated whether the groups produced different results. To do so, we used whether participants looked at the Permissions section of an app as an indicator for considering privacy. We supposed that those who looked at permissions for at least one app, in either phase, would have thought more about their model's privacy protections. In total, 29% of all participants looked at the Permissions section. Among them (Table 6.5), the differences in the number of apps installed across conditions in Round 2 are even more pronounced compared to all participants ($\sigma_{A2} = 1.314$).

The remaining participants—those who never looked at *any* permissions—are shown in Table 6.6. The differences between each phase are smaller compared to the differences between each phase among participants who considered an app's permissions when installing. Furthermore, the differences across conditions in Round 2 are smaller than in the combined population ($\sigma_{A2} = 0.478$). We also found no significant differences between the

Table 6.6: Mean number of apps installed by participants who did not view permissions ($n = 152$)

	Control	Topic	Network
Round 1	5.4	6.0	5.7
Round 2	3.9	4.8	5.0

Table 6.7: Were there significant differences in the number of apps installed *by participants who did not look at permissions* between each pair of conditions? Permutation test p-values.

	Control-Topic	Control-Network	Network-Topic
Round 1	0.17679	0.51900	0.64287
Round 2	0.18959	0.13609	0.77082

Table 6.8: Were there differences in effectiveness ratings and willingness to adopt the privacy modes between each pair of conditions? Wilcoxon Signed-Rank Test p-values.

	Control-Topic	Control-Network	Network-Topic
Likelihood	p<0.001***	p<0.001***	0.13057
Effectiveness	p<0.001***	p<0.001***	0.03606

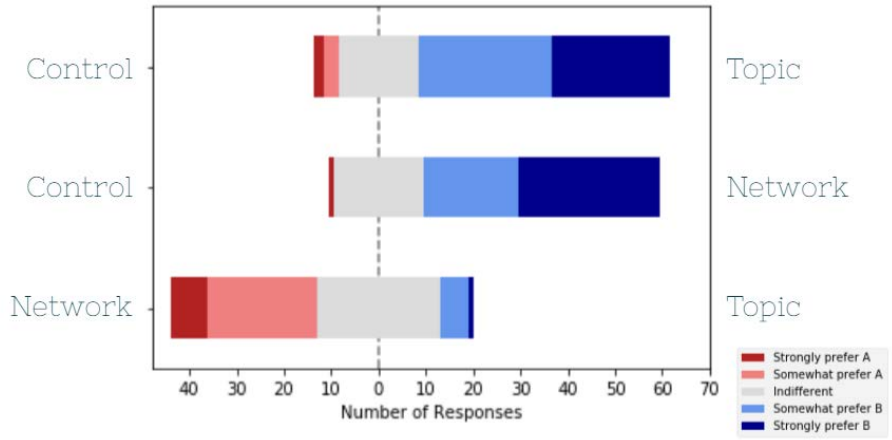
conditions in either Round 1 or Round 2 (Table 6.7).

Participants were asked how likely they were to use ALVA and how effective they found the privacy protections were after each phase of browsing the store[1]. The results of the Wilcoxon signed-rank test for these questions for each pair of conditions is shown in Table 6.8. We observed significant differences for the likelihood question for Control-Topic and Control-Network. We also observed significant differences for the effectiveness question for all three condition pairs. On average, participants found the network model slightly more effective than the topic model in controlling information being accessed by third parties (a difference of 0.25 on a scale of 5 possible discrete ratings), but this difference was not significant.

We asked participants for their preferences between the two privacy models they experienced (Figure 6.2). Most people (52 participants, 69.3%) said they preferred or strongly preferred the privacy protections from Topic to the Control (with no privacy protections).

[1]Five participants were excluded from these questions due to a data collection error.

Figure 6.2: **Preference between privacy models**: Do you prefer Condition A or Condition B?



A similar proportion (49 participants, 70%) said they preferred or strongly preferred the Network model to the control. Between Network and Topic, 48.4% said they preferred Network, 40.6% had no preference, and only 10.9% preferred Topic.

We then asked participants to explain why they prefer Round 1 or Round 2, or were indifferent. Table 6.9 lists the common themes that emerged from our qualitative coding. The majority of participants (56.5%) attributed their preference to privacy. The next most common sentiments were a lack of interest in voice assistants, a feeling that neither model offered sufficient privacy protections, and the opinion that the models were in effect the same. Only one participant explicitly stated that they were unconcerned with the possibility that their data could be obtained by third parties.

As part of our aim to understand differences between the different privacy conditions, if a participant installed different apps in Round 1 and Round 2, we asked them why this was the case (Table 6.10). A plurality (39.7%) of participants cited concern about privacy protections as a reason for installing different apps. Because of the somewhat open-ended nature of this task, however, there were a number of participants who explained that they mistakenly installed different apps in each phase, wanted to try different apps in a different phase, changed their minds about downloading an app for reasons other than privacy, or gave other answers unrelated to privacy. Of the 185 participants who did not look at app permissions when installing apps, 55 of them (29.7%) cited privacy as a reason for installing different apps.

Table 6.9: Common reasons for privacy model preferences

Reason	Count	Frequency
Prefer one for increased privacy features	121	56.5%
Not interested in these devices	24	11.2
Neither model has sufficient privacy protections	18	8.4%
Both models do the same thing	17	7.9%
Prefer one for increased sense of security	16	7.5%
Unsure of preference	10	4.7%
Don't like always-listening	10	4.7%
Can't trust third-party developers	6	2.8%
Gave other reason	5	2.3%
Prefer one for being more transparent	4	1.9%
Not interested because inconvenient for end user	3	1.4%
Both models have sufficient privacy protections	3	1.4%
Not worried about privacy	1	0.4%

Table 6.10: Explanations for installing different apps across conditions

Reason	Count	Frequency
Considered difference in privacy model	85	39.7%
Installed same apps in both stores	41	19.2%
Mistakenly installed different apps	30	14.0%
Irrelevant explanation	28	13.1%
Wanted to try different apps	21	9.8%
Changed mind about installing an app	16	7.5%

6.4 Discussion

This study's interactive app store experience provided a unique means of measuring consumer sentiment in a scenario modeling real life. Using both quantitative and qualitative analysis, we determined people's views on privacy models for always-listening voice assistants. According to Table 6.9, these generally ranged from an outright rejection of the voice assistant described in our survey to preferring one model for its increased privacy protections. Only three participants (1.4%) responded that they believed there were sufficient privacy protections in place for both models they were assigned to, indicating that neither of the models are good enough by themselves for most people to be comfortable with them.

The results of this study demonstrate that, as a whole, people are generally concerned about the privacy protections, or lack thereof, offered by always-listening voice assistants. This holds true despite the fact that these models may be too simplistic or incomplete; perhaps, users may have the perspective that something is better than nothing. Our findings show that consumers do seek to make choices to protect their privacy when considering new technologies, as demonstrated by the number of apps they installed, and is reinforced by explicitly inquiring about their consideration after browsing the store. Prevailing sentiments from our qualitative analysis show concerns about malicious third parties gaining access to sensitive data.

Do people install more apps when there are privacy controls?

We wanted to determine whether participants would install more apps for their voice assistant when given some form of privacy protection for the speech that the voice assistant would be able to share with third-party developers. We hypothesized that users would install significantly more apps when there was some privacy model in place. Our results provide some evidence in support of our hypothesis, as the participants who received either the Network or Topic conditions installed significantly more apps than the Control condition in Round 2. On average, participants in Round 2 installed the most apps for the Network condition, then the Topic condition, and finally the Control condition.

To understand our participants' behavior on a deeper level, we reran our analysis of the mean number of apps installed after separating participants into two groups—those who looked at permissions and those who did not. We reasoned that those who did consider app permissions would show a larger effect, and this was true. Despite making up the majority (71.0%) of our participants, those who did not consider the Permissions section when installing apps showed no significant effect in their data. Those who did were more careful with the number of apps they chose to install, as shown by the mean number of apps installed in Table 6.3.

Do privacy controls improve the perception of the assistant?

We investigated how likely participants would be to use ALVA for each condition. The results support our original hypothesis that privacy controls make people more willing to use always-listening assistants, as participants were significantly more likely to say that they would use ALVA under the Network or Topic models than in the Control model.

We also investigated how effective participants found each privacy model. Again, the results support our original hypothesis, as participants thought they could more easily control the information they shared with ALVA under the Network and Topic models.

Which privacy controls do people prefer?

With regard to the question of which of the two privacy protection models participants favored more, participants found the Network model to be more effective than the Topic model. However, they were not significantly more likely to say that they would use the Network model over the Topic model, and the mean number of apps installed was also not significantly different between the two. This similarity indicates that neither model is robust enough by itself to be a clear winner. A much more complete and granular solution is no doubt required to increase device functionality while protecting user privacy.

Why are differences only present in Round 2?

A question that arises from looking at the results concerns why the differences were significant in Round 2 but not in Round 1. We have no certain answer, but we hypothesize that there may be a couple of factors contributing to this discrepancy. The nature of our survey design, being more open-ended, may cause participants to have interpretations of the task that are different from our original intention. For instance, our qualitative analysis shows people installing different apps in the two phases because they wanted to try different apps, or other reasons that are not relevant to the privacy model. Perhaps when initially given a privacy model, participants are more willing to install apps because of the novelty of the technology, but consider privacy more when they learn about alternatives.

To investigate this difference, future studies can ask more directed questions relating to why participants installed different apps for each device. Furthermore, another question that follows is whether participants who cited privacy in their responses showed different behavior relating to the number of apps installed or whether they looked at the permissions for the apps that they installed.

Limitations

The two-phase design of our survey may not be the most ideal means of gathering data. This is because participants may be fatigued by the second round, as they are asked to repeat a task, and their behavior might not always be due to their consideration of the privacy model. For instance, one participant responded that they had forgotten what the details of the first privacy model they encountered, which may affect their preference, likelihood, and effectiveness ratings.

While we had attempted to create a more realistic task for participants to complete in our survey, it is still not entirely realistic. One consideration is the absence of a more complete privacy solution in our survey. People familiar with smart speakers like Amazon Alexa or Google Home may be aware of a suite of existing features on these voice assistants, such as light-up indicators to tell when the voice assistant is listening, a means of deleting data, guest mode, and a mute button. Despite being capable of always listening, future devices

will likely still have the option of requiring a wake word. Also, we listed an anonymous third party developer as the creators of the apps in the app store. In reality, consumers would want to make sure they could trust the companies behind the apps they install. The creator of the voice assistant would likely vet the apps in their ecosystem, and consumers could view more detailed information about the developer on an app's store page, but we eliminated these variables for the purposes of this study. While we attempted to use a real-life scenario to focus on determining consumer opinions of privacy models, participant opinions were undoubtedly colored by their knowledge of the current voice assistant ecosystem.

6.5 Future work

One natural next step is to incorporate more privacy models into the study. There are many such models that could be tested, and eventually we could determine which are most favored. Models can be combined to create a stronger, more complete privacy solution. We can also focus on individual models to develop and formalize them in concert with consumers.

Participants' stated opinions differed from their expected behavior when installing apps and looking at app permissions. When it comes to participants' responses, the most common sentiments do show concern about how their privacy might be violated, but the majority of participants did not look at app permissions. We could analyze participants on an individual level rather than in aggregate and investigate why this is the case. This could take the form of another free-response question in future iterations of this study.

6.6 Conclusion

We designed an interactive voice assistant app store experience to use in concert with a traditional survey. Our findings show that participants as a whole installed more apps and indicated preference for voice assistants with privacy models. On a more detailed level, however, despite over half of participants indicating that privacy was a reason for preferring one of the two conditions they were assigned, the majority of participants did not look at the permissions for the apps they installed.

Chapter 7

User Perspectives on Runtime Permission Requests

To find out how people would react to different kinds of runtime permission requests, we had participants in this study hold conversations while getting ambient suggestions (and plenty of permission requests) from a passive listening assistant, which we simulated in real time using the Wizard of Oz technique. Most of our participants were excited about passive listening, but wanted control over the assistant's actions and their own data. They generally prioritized an interruption-free experience above more fine-grained control over what the device would hear.

> I think that people already know we're moving towards a technology-based world, but they've also seen the negative sides of that. So I think something that is in your household—that you are the owner of—you would want things to run smoothly according to what you are comfortable with.
>
> Participant 10A

7.1 Introduction

Passive listening assistants, like just about any system that incorporates asking for permission, have a fundamental choice about the timing of such requests. Should they happen at the point in time where data access is needed? Or is it better for them to take place earlier, for example when the app is installed? (It is also possible to defer the permis-

sion requests altogether, making them subject to after-the-fact review—an option we will explore in Chapter 8.)

Previously, in Chapter 6, we explored the approach where users reviewed permissions at install-time. We found that people demonstrated a strong interest in privacy controls, but a high fraction of participants did not pay attention to the permissions of the apps they were installing. This result is consistent with the literature, which has universally found limited efficacy of install-time permissions [74, 76, 185, 224].

The ineffectiveness of install-time permissions has motivated research and adoption of alternative approaches. On smartphones in particular, asking at the time of data access (i.e., runtime permissions) has become the preferred alternative [90, 41]. Doing so helps provide the user with greater context about data requested by the app as well as (potentially) the reason for the request.

How would runtime permissions work in a passive listening context? This chapter focuses on exploring this subject. Specifically, this study's primary research question is: **what should be the user experience of runtime permission requests in a passive listening assistant?** The remainder of this chapter presents our exploration of this topic. To begin, we need to specify in more detail what the permission requests are doing (§7.2) and then how to best evaluate the user experience (§7.3).

Architecture Runtime permission requests are broadly compatible with a number of the privacy models discussed so far (§5.8). All of them could therefore be valid subjects for the experiments in this chapter. However, evaluation is best done with (some) one specific model, and for that purpose we chose *network-restricted mode* (§5.8). Under this approach (Figure 7.1), third-party applications gain full access to all audio, but run completely sandboxed from the outside world. While they are allowed to make network requests (as needed to send or receive new data), they must be reviewed by the user, after being transformed into a human-readable form. These are the permission requests that are evaluated in this study.

As an example, a weather app, running offline by default, would be able to decide for itself whether some speech is relevant to it. Once it decided that action was merited, it would need to communicate with its API, over the network, in order to actually obtain up-to-date weather information. To do so, it would need to generate a permission request, by providing the speech it wanted to share, along with a pre-registered template. The platform would show the resulting permission request to the user and, if approved, send the speech in question to the app's servers.

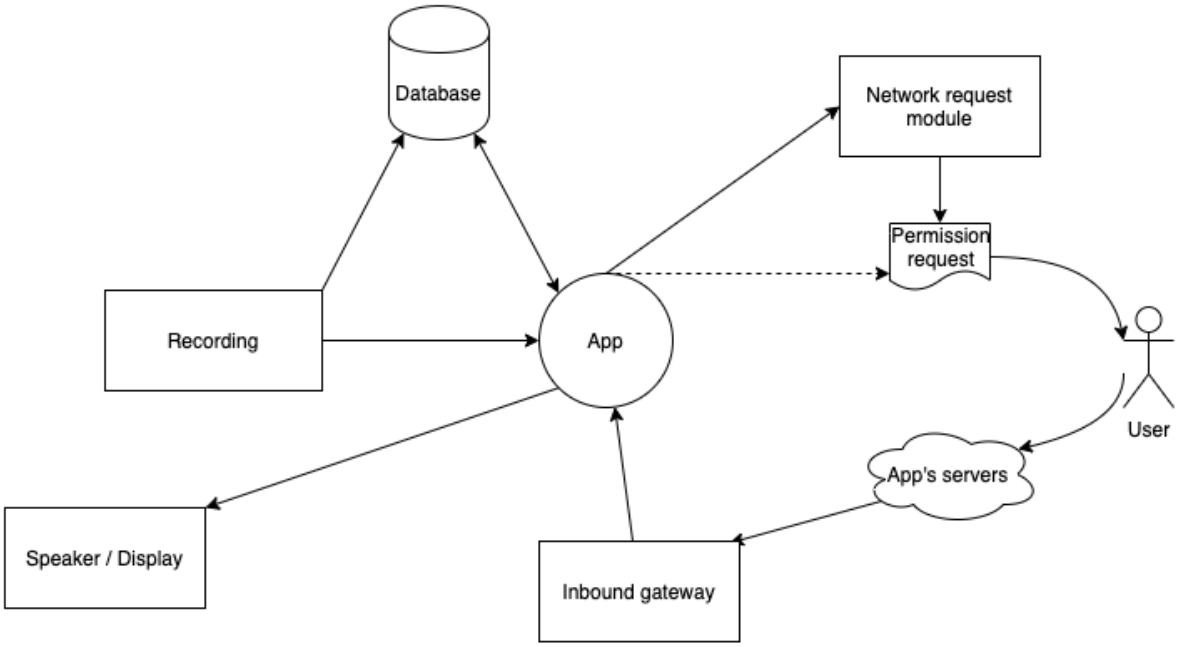


Figure 7.1: **App and assistant architecture**: the data flow in the passive voice assistant platform that enables the permission requests in this study.

7.2 Design of runtime permission modes

According to our architecture assumptions, a runtime permission request for a passive listening app happens as part of the following sequence of events:

1. The user says something. (“I hope it doesn’t rain while we’re in Hawaii”)
2. The app decides this speech is relevant to it. (Per our architecture assumptions, this happens in a sandboxed module.)
3. The app identifies the information it wants to share over the network. (In this scenario, the weather app would want to share the location, Hawaii.)
4. The user is then shown the permission request. (“The weather app would like to share ‘Hawaii’ over the network, allow or deny?”)
5. If the user approves, the requested information can be shared with the app over the network.

Even if we decide that our system has runtime permission requests, and assume the architecture that yields the flow above, there is still a large design space to specify how exactly

this figures into the user experience [70]. One of the main questions at issue is: should every attempted data access generate a user-visible permission request?

Ask every time

One option is to ask the user every time an app wants access to a sensitive resource. The benefit of this approach is that it gains the user's positive assent on every invocation and guarantees that a human reviewed every permission request.

The downside of asking for every access request is that it places a very high burden on the user—one that is very likely unreasonable. For example, researchers found that apps on a single smartphone generate over 100,000 requests a day to protected resources [224]. Reviewing each one would certainly be untenable.

There is some possibility that a similar manual approach to passive listening permission requests would not be quite as impossibly onerous. To come to this conclusion, we considered all the sample always-listening apps and scenarios from Appendix B and enumerated the permission requests each one would generate. We found that most use cases would generate only one or two permission requests per conversation. Unlike with smartphones, these numbers makes it potentially feasible for a user to address each request themselves.

While the projected number of permission requests is at least relatively promising, it does not necessarily make asking every time a good idea. First, in our analysis, some apps and scenarios *did* result in a large number of permission requests in a short amount of time. For example, going over a shopping list might result in a permission request for each individual item (though careful programming of the app could avoid this). Regardless, if many different apps are activated over the course of a conversation, or over the course of a day, this can still add up to a large amount of permission requests that a user would have to review.

Frequent or repeated permission requests will likely annoy the user and are very likely to result in habituation, a result well-studied in the literature [213, 212] and also known as warning fatigue [8]. Thus, while we allowed for the possibility of this approach working, our hypothesis was that ask-every-time runtime permission requests are too burdensome for most users.

Ask on first use ("*Rules*")

Because asking every time is impossible, and the limitations of install-time permissions are by now well-known, smartphone permission systems have turned to another way of getting user approval: ask on first use.

As its name implies, ask on first use works by asking for the user's permission the first time an app needs access. The user's decision is then applied to all future permission requests. In essence, the user's options are to "always allow" the app or "always deny" it.

Some variants of this design have emerged. For example, Android has added an "allow only this time" choice. This has the benefit of allowing users to put off their decision if they are unsure. Offering this option also allows sufficiently motivated users to revert to the ask-every-time mode.

How would ask-on-first-use work for passive listening apps? The most literal adaptation from the smartphone model would see the app asking to access the user's speech the first time the app thinks it is relevant. If the user grants this permission (always allow), from that point on, the app could access that resource whenever it wanted. In our passive listening architecture, this would mean it has carte blanche to send any speech over the network.

This approach is liable to abuse. An app could make an appropriate permission request the first time around; then, in the future, it could make requests—and have them automatically be granted—at inappropriate times. It is worth mentioning that this exact abuse possibility exists for smartphone apps as well [190], but the ask-on-first-use permission mode has been quite successful regardless.

Nonetheless, we felt that a higher degree of restrictions would be appropriate for a passive listening app since it has access to all of a user's speech, while a smartphone's app permissions are scoped to a specific resource (e.g., camera, location).

Instead, we proposed a design that scoped an app's access to a specific type of speech or entity, for example:

- Always allow the weather app access to locations
- Always allow the events app access to dates and times
- Always allow the supermarket app access to groceries

We refer to this as the *Rules* design and believe that it offers a considerably more effective constraint on the ability of an app to listen inappropriately.

Can these *Rules* be formulated for every app? To answer this question, we again analyzed each use case in our bank of sample apps. We considered whether the permission it requested referred to something that could plausibly be classified into a single entity, group, or type. Examples of this include (as above) locations, date, numbers, types of speech, categories of physical objects, emotions, etc.

We found that a majority of our sample apps were amenable to the *Rules* design. However, not all were: apps where the target speech (or part of it) were completely open-ended may not be a good fit for the *Rules* design. Examples include action items for a to-do list (though a hypothetical constraints for these could be literal actions or just verbs) or meeting topics for a scheduled calendar event. However, we observed that these types of topics come up infrequently enough that reverting to the ask-every-time mode for them—while using *Rules* for the rest—would still meet our objective for not interrupting the user too much.

Contextually relevant permissions ("*Learning*")

Even if the *Rules* design could be applied to every app—that is, we could come up with a narrowly targeted entity type that effectively constrained listening for any use case—there might still be a fatal flaw undermining these permission requests: they might not be contextually relevant.

As we know from the theory of contextual integrity (see Chapter 3), people might have varying preferences even when a variety of factors are kept constant—such as the data type, subject, and recipient—because of different contexts. This is especially likely to happen when the information in question is speech in a conversation. If the weather app uses a location to look up the weather when people are talking about traveling, this can be helpful; if it does so while they are discussing their secret getaway plans, likely less so. As this example illustrates, contextually inappropriate data requests may happen if the app is not malicious, simply because contexts are varied and can be nuanced.

One answer to this is to go back to asking the user about every permission request, so as to let them make a decision based on the correct context. We have already discussed, however, why this may not be a good idea (§7.2).

Another approach is to attempt to differentiate between the different contexts automatically. The idea was first developed by Wijesekera et al. for smartphone permissions [225], where they collected a variety of contextual factors and then used them and machine learning to predict whether people would allow or deny permission requests. This approach is further inspired by work on personalized privacy assistants for mobile app permissions [130] and other smart home devices [60, 52].

While the contextual factors for a passive listening assistant will be based on conversational context and thus very different from those for smartphone permissions, we envision an analogous approach where the assistant's platform (trusted in our threat model) could determine, based on the context of a given conversation, whether an app's permission request should be approved or not. We refer to this as the *Learning* design.

We will leave the exact details of this approach's implementation unspecified, as we be-

lieve that they are unlikely to be feasible with today's state of the art natural language processing. But we conjecture that, after advances in NLP, something like this will conceivably be plausible, after some amount of training and supervision from the user, and will make use of some form of machine learning.

Challenges of this approach include the relative immaturity of modern machine learning systems and natural language processing and the fact that we do not know the features necessary to identify and distinguish different contexts. This makes for a potentially promising avenue for research; however it does somewhat call into question the practicality of this design.

Trade-offs between permission designs

To summarize the proposed permission designs and their relative benefits and drawbacks:

- Ask-every-time ensures user's awareness and (ostensibly) consent for each request but likely makes for a user experience that is too annoying (even if it seems more practical than in smartphone permission systems).
- Ask on first use (*Rules*) reduces the number of requests a user has to deal with but may not be applicable to all apps and may let through requests because people's preferences differ across contexts.
- Permissions that take advantage of *Learning* may be able to distinguish privacy contexts but their success relies on advances in natural language understanding, which are not guaranteed to come to pass.

Setting aside the practicality of each approach, our goal in this study is to understand each of these permission designs from the perspective a user. **What is the user experience of each permission approach?** What are their relative advantages and disadvantages? Which are most likely to be **acceptable for day-to-day use** and which **enable people's trust**? This study is dedicated to answering these questions.

7.3 Methods

This section describes our approach for answering the research questions about the user experience of runtime permissions. We begin by explaining the reasoning behind our experimental design, then convey our procedures in greater detail. We conclude this section with information about our participants and how they were recruited.

Study design

Prior work on permission requests primarily focused on smartphones [74] and other existing systems. The researchers were therefore able to deal with real devices, for example by instrumenting users' interactions with them [225] or constraining them to behave in ways necessary for a laboratory experiment. Because we cannot do the same with a hypothetical device, our first task is figuring out *how* to do this research.

Simulating an assistant

To answer the research questions of this study, we need to get people's opinions about the user experience of the different permission modes. We could get them abstractly (e.g., through a survey), but their opinions would be more authentic if they could experience the interfaces directly.

Since passive listening devices do not exist, we could make our own; however (as elsewhere in this project) the limitations of current technology and our resources mean that our implementation would not actually work well as an assistant, which would necessarily skew participants' perspectives.

Instead, we decided that the best course of action would be to simulate the experience of a passive listening assistant for our participants. This is known as the Wizard of Oz technique and is commonly used in user experience research [107, 59, 141, 179]. When the Wizard of Oz technique is used, the actual product is not implemented, but the interface and interactions with it are simulated for the benefit of the potential user, with a researcher performing the actions expected from the software.

The requirement that a researcher manually simulate the software features necessarily limits the scale of a Wizard of Oz study. However, this aligned well with our goals, as we were interested in open-ended qualitative feedback to collect rich insights from participants, rather than more structured quantitative measures, which would not be as helpful for answering our research questions. Thus, we settled on a qualitative user study, employing the Wizard of Oz technique, to present the passive listening assistant to participants and get their insights on the potential permission modes.

Another consequence of having a researcher simulate the software is that the simulation must be constrained to a specific time and duration, as opposed to just giving people an assistant and having them interact with it on their own time. This meant that, for our study, we would have to recruit participants in groups of at least two and explicitly direct them to have a conversation, with the (simulated) device listening.

Structuring the experiments

Having decided that study participants would experience the different permissions variants, the next major decision we faced was whether to conduct a between-subjects or within-subjects experiment. With a strict implementation of the former, we would assign participants to exactly one permission mode and compare their experiences and feedback to those from other conditions. While such designs typically offer greater ecological validity, they are better suited for quantitative comparisons, whereas we are looking to collect more open-ended feedback, as discussed above. In contrast, exposing people to multiple permission modes, in a within-subjects design, allows participants to reflect on the differences and express their preferences. Because of the opportunity to collect this sort of in-depth data, we chose this structure for our study.

Procedures

On a high level, the study was divided into three activities:

1. Explanation
2. Interaction
3. Interview

This cycle of activities repeated three times over the course of the study.

More specifically, the study began with an explanation of (existing) smart speakers and intelligent voice assistants, followed by an introduction to the novel concept of passively listening and the behavior of the assistant the participants would be interacting with that day. (We named our assistant "Alva." Participant quotes later in this chapter refer to the assistant by this name.)

Our introduction included a demonstration of the "features" of the assistant as well as the runtime permissions that are the focus of this study. It emphasized the following points:

- The always-listening nature of the assistant
- Its ability to offer suggestions based on conversations and queries not directed at it
- That all features are enabled by apps, which are created by third-party developers rather than the manufacturers of the device
- The requirement for apps to ask for permission before accessing speech

After participants learned about the passive listening voice assistant, they took part in an interactive session where they engaged with the assistant and its permission system.

Interactive component

The core idea of the interactive session was to simulate a passive listening assistant for our participants. Since the assistant is supposed to be primarily passive, it reacts to people engaged in their own conversations (rather than addressing it directly). To support this, we recruited participants in pairs, so that they would have someone to talk to. Then, during the interactive session, we asked the pair to have a conversation, while the assistant tried to offer relevant suggestions.

Wizard of Oz implementation

Our study was conducted remotely, over a video call. For the interactive portion of the study, the interviewer shared their screen, which contained a browser window showing the presentation view of a rapid prototyping tool.[1] During the interactive session, when participants were talking to each other, the interviewer would update the screen, as quickly as possible, based on what the people were saying.

The content on screen would either be a permission request or (after permission had been granted) information relevant to the current discussion topic. Examples of the latter included facts about weather, tourist information, ticket prices, etc. This was accomplished by the interviewer entering relevant keywords into a search engine, taking a screenshot of the summary boxes returned, and pasting the screenshot into the prototyping tool, to be seen by the participants. For the permission requests (discussed in greater detail below, §7.3), we had pre-made templates for each app, which the interviewer updated with speech from the participants, then brought into the viewport.

Due to the manual nature of the simulation, there was an average delay of approximately 5–25 seconds between when participants said something and when the corresponding visual appeared on screen. We warned participants about this delay upfront, and while many commented on it, others found it acceptable even for a real system.

Three rounds

Since we decided that our study would be a within-subjects experiment (§7.3) to test the three different permission modes (§7.2), each pair of participants went through three interactive sessions. The first interactive session lasted five minutes and always featured the ask-every-time permission design. It served as a practice session for participants to become familiar with the assistant and its behavior. Subsequently, participants went through two more sessions, each 10 minutes long, testing the *Rules* and *Learning* designs. (The order of these was randomized.)

[1] https://www.figma.com[2]

[2] Thank you to Dylan Field, Evan Wallace, and others for making this tool.

Task selection

To demo the assistant's functionality, we needed to have people talk to each other, and the assistant needed to provide relevant suggestions. We chose to provide participants with conversation prompts, one for each of the three interactive rounds. This served a dual purpose. First, it helped participants, since we believed some might struggle with a completely open-ended prompt ('talk about anything'). More practically, it focused conversation topics on subjects where the assistant (as simulated by our Wizard of Oz implementation, i.e., a researcher making search engine queries) could plausibly offer relevant suggestions.

To that end, the three conversation topics we chose were:

- Cooking dinner
- Arranging weekend plans
- Planning a vacation

For each of these conversation topics, we came up with a list of passive listening apps that would be active for that conversation, for example *Recipes* and *Shopping List* (for cooking) and *Flights* and *Weather* (for making plans). The complete list of apps for each scenario can be found in Appendix C.

Permission designs

A major design consideration was how participants would experience the permission prompts. The primary choice would be whether they would be presented visually or as a voice prompt. We opted for the former, primarily because they are less likely to interrupt the flow of conversation, allowing those speaking to finish their thoughts before addressing them. However, this approach has limitations, especially in a more realistic setting where users are not in front of the assistant's screen during their conversation. We therefore included this design choice as one of the discussion topics in our interview guide (C), to collect people's views on the trade-offs.

We came up with a design and behavior pattern for each of the permission modes (§7.2). The default permission design was a dialogue prompt, stating "*App name* would like to *perform action* with *participant's speech*"; for example, "*Weather App* would like to *look up the weather* in *Berkeley this weekend*" (see Figure 7.2). The dialogue showed two "buttons", *Allow* and *Deny*, and participants were instructed to state their preferred action (i.e., say one of those two words out loud) for it to take effect. True to the name of the ask-every-time design, in the corresponding phase, such a dialogue was shown every time the participants said something the assistant could act on.

Figure 7.2: **Sample permission request**

In the round where *Learning* was tested, the same dialogue design was used, but it was shown only once or twice for each app. Any subsequent lookups were performed without a permission request to precede them, as a simulation of the assistant having "learned" the user's preferences.

The *Rules* variant permission request had a slightly different design. Besides the options to *Allow once* and *Deny once*, it featured two additional choices: (using the weather app as an example) *Always allow* Weather App *access to locations* and *Always deny* Weather App *access to locations* (see Figure 7.3). These options were adjusted for each relevant app and data type (e.g., *Always allow Calendar access to dates*).

Figure 7.3: **Sample permission request for *Rules* design**

As part of our explanations, we told our participants that both the *Rules* and *Learning* designs had an extra feature: the "review mode." The intention behind it was that users should be able to see what decisions were made automatically on their behalf, and change them if necessary. Participants could invoke this mode during the simulation by asking to review their permissions. If they did so, we showed them a separate screen that contained copies of approved or denied permission requests. One of the research questions we had was whether participants would make use of this feature.

Misbehaving apps

Most apps in the simulation were intended to perform correctly, only asking permission for relevant information at relevant times. However, we also wanted to see how people would react to inappropriate permission requests. This would also serve as a basic test of the permission system's effectiveness: if people identified and denied inappropriate permission requests, this would suggest that they might similarly be able to catch and prevent actually malicious apps.

To simulate an inappropriate permission request from a misbehaving or malicious app, the researcher running the simulation would choose a random point during each of the three interactive sessions to show a permission request from a new, previously unseen app, which would request access to the last thing said by the participants, even though it had no relevance to the app's actual functionality. The three misbehaving apps were *Celebrity gossip*, *Bedtime stories*, and *Smart bulb*.

Interview questions

After each interactive session, we interviewed the pair of participants about their experience. The interview focused on a few key aspects. First, because passive listening assistants are a novel concept, and because this is the first study in this dissertation where we were able to collect open-ended responses about them, we asked participants about their general impression of passive listening and its privacy implications in particular. The next category of questions was general questions about permission prompts, such as their clarity, modality (visual versus audio), and how much trust they engendered. We also collected perceptions of the different permission modes, as well as people's preferences between them and the reasons behind these. Finally, we asked directly about privacy with respect to the passive listening assistant, including any concerns people had and controls they wished to see in a device. The complete interview guide can be found in Appendix C.

Analysis

We analyzed the interviews in our study using an inductive approach to thematic analysis [34]. A single coder reviewed each interview and created a codebook, consisting of the set of themes identified across participants' responses. On subsequent passes through each interview, they were annotated with each theme from the codebook that was present in the response. Finally, illustrative quotes were selected for each theme, to be presented in §7.4.

Recruitment

We recruited participants for our study by advertising a "computer gig" on Craigslist. Those interested were invited to fill out a screening survey, which asked for basic demographics and three free-response questions about the respondent's use of intelligent voice assistants, smart speakers, and other smart home devices. From those who filled out the screening survey, we invited participants to the main study, targeting approximately equal numbers of people with different levels of experience with voice assistants and smart home devices: low (limited or no usage of voice assistants), moderate (usage of smart speakers only), and high (multiple smart home devices besides smart speakers). We also aimed to balance our sample demographically.

The study session lasted 90 minutes, and participant pairs received $60 in compensation (to be shared by the two people). Our study's procedures were approved by our Institutional Review Board.

Demographics

Our screening survey was completed by 176 people, from whom we selected 23 pairs to participate in the study. The majority of the pairs (52%) consisted of spouses or partners, 30% were made up of family members, and the others were friends or roommates (9% each). Among the 46 participants in our study, 57% were female and the mean age was 37.

7.4 Results

This section describes participants' behavior during the interactive sessions and reports the major themes that resulted from analyzing the interview portions of our study.

What did people think of passive listening?

Before diving into the details of the permission systems, we wanted to get a sense of people's general perceptions of passive listening, since one's attitude about the concept typically colors their impressions of its features. When making sense of the passive assistant's functionality, existing smart speakers were a baseline for feature comparison: *"It just seems like an enhanced Alexa"* (P16B). We found that our participants were, on the whole, receptive and even enthusiastic about the idea of a passive listening assistant when it was first introduced to them.

> *"I think it's nice that you don't have to call out the name because it's already picking up on the conversation"* (P16A)

> *"Having a virtual assistant in the same room, just kind of like answer my thoughts even if I didn't know about it, that's obviously a very great convenience to have."* (P18A)

After experiencing the passive listening assistant as part of the interactive sessions, participants were even more keen, exclaiming *"It was really, really cool"* (P19A) and *"It was dope"* (P19B). Others commented that the device overcame their initial skepticism: *"I was surprised at how good it was"* (P5B).

One of the closing questions in our interview was whether the participants would choose to adopt a passive listening assistant. With only a few exceptions, our participants agreed that they would.

However, just because the concept of passive listening received positive feedback from our participants, did not mean they were unaware or unconcerned about the privacy implications of such a device. For example, a number of participants relayed stories of existing devices listening at unexpected times, such as an Alexa interrupting a conversation to answer a question no one asked. Such accidental activations remain a regular occurrence, as we know from Chapter 4 and other research [65, 186].

How did people feel about passive listening and their privacy?

When we introduced the concept of passive listening and asked participants for their initial reactions, only a small fraction mentioned privacy. However, over the course of the subsequent conversation, we heard a variety of nuanced viewpoints and concerns. This section explores these in detail, as we feel that an understanding of these considerations can aid in the design of appropriate privacy controls.

Privacy nihilism Only a small number of people claimed that they did not care about privacy at all, resorting to the common trope about having nothing to hide:

> *"I think we're very average people, you know, and privacy is not an issue, at least for us."* (P4B)
>
> *"I guess I don't have too much to hide."* (P22B)

Resignation A more common opinion, though still in the minority, was privacy resignation, a phenomenon that has been observed in other contexts as well [187]. While these people valued their information, they felt that attempts to protect it would, to a large extent, be futile. One of the reasons for this was that modern technology itself is designed to collect as much data as possible:

> *"In this day and age, everybody's recording everything."* (P21B)
>
> *"We have technology everywhere, like that's kind of beyond us at this point."* (P23A)

The other common reason for resignation was the existence of hackers and belief in their ability to obtain almost any information:

> *"Anybody can hack into anything."* (P17A)
>
> *"There's always third parties out there now. If they really want to hack in anything it's easy–so easy–for them."* (P12B)

Worries about hackers were common among other participants, even if they did not express quite such an absolute conviction about hackers' abilities. As evidence, those who talked about this subject cited recent high-profile cyberattacks that had been reported in the media. P1A, for example, felt that the government was powerless to stop these (*"they can't secure nothing"*).

Privacy contradictions Even the people who claimed that they were not concerned about privacy actually demonstrated nuanced views. (This is consistent with much research on the so-called "privacy paradox" [196].) For example, P9B described themselves, *"I'm pretty much an open book. I mean, I think a lot of people worry too much about privacy."* Yet, shortly thereafter, they provided an explicit example of data types they did consider private: *"If I ask [my partner] for a social security number, if I'm filling it out, you know, I may not want [the assistant] to do things like that."* P17A drew a clear distinction between two privacy-invasive behaviors, one that they did not mind and another they considered unacceptable:

> *"Privacy... I don't really care that they're kind of tracking me in a way, but I don't want someone to break into the system and find out where I am and stuff. That's scary."* (P17A)

Privacy concerns The majority of our participants articulated some privacy concerns about passive listening devices, either organically, over the course of the interview, or directly, when prompted in one of the final questions. Often, these concerns were attributed to "some people," rather than themselves:

> *"I think this would be something that I feel like a lot of people would be concerned about."* (P18A)
>
> *"Some people might might not like [passive listening]."* (P7B)

> *"Already they're concerned that the other voice assistants like Alexa, it's always listening even if you don't say their wake-up command. People are concerned that they're always listening. But this right here, they* know *that it's always listening. So it might raise some concerns for those people that are already squeamish about the voice assistants."* (P9A)

What were people concerned about? Only a couple of interviewees expressed discomfort with the always-listening nature of the device more generally. (P1A, for example, referenced Orwell's *Nineteen Eighty-Four*). On the whole, though, always listening did not bother people; instead, there was specific information and scenarios that they were concerned about.

Sensitive data types A common concern among participants was that the assistant would capture specific types of information, which they preferred for no one to be able to hear. Consistent with popularly held notions about what is considered private information [168, 40], the most common data type participants worried about was financial information, such as bank accounts, credit card details, and social security numbers. We also include passwords, which some mentioned, in this category.

> *"Anything that has to do with my banking information, anything about money."* (P16B)
>
> *"Like your address, your social security number."* (P15A)
>
> *"I'm talking to customer care and they ask me for my credit card details or my PIN."* (P20B)

Participants were also worried about the device overhearing conversations on subjects they considered sensitive.

> *"What if I'm talking to someone, you know? We're planning a funeral or something? Maybe I don't want Alva listening. And maybe that person is sharing stuff and they don't want it listening"* (P8A)

This quote also demonstrates concerns about bystanders and visitors—non-owners of the assistant whose voice might be captured against their will, a theme we first observed in Chapter 4.

A few pairs mentioned that they would not want an assistant listening if they were gossiping.

> *"Let's say we're gossiping."* (P19B)
>
> *"We gossip about people and different things."* (P8A)

While medical information is often considered sensitive in the United States [168, 54], only two participants brought it up in our interviews as something they did not want an assistant to hear.

> *"I wouldn't want the whole world to know my medical history."* (P1A)
>
> *"When it comes to financial and medical things, that should obviously be protected."* (P19A)

Arguments or disputes were another example of a specific sensitive conversation subject, which a few different participants remembered.

> *"We got into an argument and we're going, 'he said, she said.' "* (P7B)
>
> *"If we're ever having, let's say, an argument. Or we're, you know, having a tough conversation or something."* (P4A)

In the latter case, however, the interviewee felt that there actually could be a role for a (sufficiently smart) assistant to step in and serve as a mediator: *"If it would say, hey, take a break. You two should take some time apart right now."*

Other examples of sensitive conversations that participants came up with included *"family matters"* (P2A), relationship and cheating— *"I'm having an affair with somebody"* (P1A)— and business calls made while working from home:

> *"Now it's work from home, or I might be just calling a colleague and talking. And if the machine records it, I wouldn't want it. Not that anybody else would listen to it, but that's confidential."* (P20B)

While most concerns focused on specific data types, such as the ones above, one person brought up the issue of metadata leakage, pointing out that even innocuous conversations could reveal potentially sensitive details. They felt, therefore, that all data—not just "private" conversations—merited protection. Indeed, a variety of inferences can be made from voice even without considering content [117], and advertisers have sought to exploit all information available to them [145].

> *"I think anything, everything should be protected, if it's not my voice. Because anything can be used. Like me making a dinner reservation for seven o'clock is not a problem, until the stalker breaks into my house and wants to find out what I'm doing at seven o'clock. So it could be information that's not harmful. But in the wrong hands, it can become harmful. I know I'm using an extreme example, but things are weird and weird things can happen."* (P19A)

Data uses Some of the concerns voiced by participants focused on what would happen with their information, rather than the specifics of that data. Specifically, a number of people expressed discomfort with the possibility of their data being sold. Most discussed this threat abstractly, without reference to any specific entities who might be buying the data and how it might be used, but at least one participant explicitly did not want their data to be used for advertising.

> *"Your data is secure: it's not some person who is taking your data and selling it to somebody."* (P20B)
>
> *"If they were selling my information and then if I was wanting to plan a trip to Hawaii and then suddenly I received calls from my travel agent or something."* (P14A)

Intra-household data leakage When discussing scenarios they were concerned about, several participant pairs brought up the possibility that the passively listening assistant would overhear conversations and would later reveal their contents, in one way or another, to other members of the household, leading the person to find out secrets others are keeping from them.

> *"Maybe something that you discussed—it was really really private—popped up on the screen and somebody else in the house saw it"* (P6A)

Secrets, of course, need not be a sign of malfeasance or problems in the household. While one participant (P1A) jokingly mentioned the scenario of having an affair, others conjured more benign examples:

> *"Kids, they're very nosy, so they don't need to know everything. What if you're planning a surprise party and they're going to want to be, like, oh what were mom and dad talking about?"* (P10A)
>
> *"Let's say I'm throwing a surprise dinner for [partner]. [...] But then [partner is] home and [assistant] just starts blurting out next week's plans, and I'm, like, did I freaking tell you to do that?"* (P19A)

Impactful actions Overwhelmingly, the concerns expressed by participants in our study focused on impactful action the assistant might take. These worries—that the assistant would do something they would disapprove of—were much more common than about what would happen with their data. We heard these both when people first learned about passive listening and after interactive sessions, when they had become familiar with the apps in our study and their capabilities.

While different in kind, the *contexts* in which these concerns appeared were similar to those for the data types above. For example, the top concern was that the assistant would take actions with financial consequences, such as buying items or booking tickets.

> *"I want to make my own financial decisions."* (P1A)
>
> *"If it's something that's going to charge me…For example I'm okay with requesting an Uber via a smart assistant, but I just want to make sure that everything is correct. For example if I'm requesting an Uber in an hour or at a certain time instead of right now, that's a big distinction. So just the clarification of 'okay, did you want that Uber to get to you right now or do you want it at X time?' "* (P4A)

People also worried about social consequences that might follow from the assistant performing actions without approval, for example messaging friends or creating invitations. (Social communications are often a source of privacy concerns [195, 16].)

> *"I would always have it [ask me] only when it's going to send something to someone else, like a person in my contacts or something else."* (P4A)
>
> *"If I said to him like, 'oh I need to call my mom' would it, like, call my mom?"* (P14A)

Even if the assistant's actions affected no one but the user of the device, participants observed, they are still able to cause annoyance or inconvenience, for example through unwanted events being scheduled or alarms being scheduled.

> *"If you're having a discussion with someone and it comes up, hey, should we cancel dinner for tomorrow? […] She might automatically do that without hearing the end result, or put random things on your calendar."* (P9B)

While the inconvenience stemming from such autonomous actions may be judged as relatively minor, participants often felt that it was these violations that permissions ought to be (or were) guarding against.

How well did the runtime permissions work?

A major goal of this study was to observe how runtime permissions would perform in a semi-realistic setting. Overall, we observed that nearly all participants understood how to use them right away (following the brief initial explanation): when, during interactive session, permission pop-ups appeared, participants addressed them knowingly and with purpose. The majority of permission requests in our study were approved; when participants denied one, it was typically because they considered the service unnecessary, for example if the assistant offered to show driving directions to a destination the person knew well.

A few pairs struggled a bit more during their first interactive experience. When permission requests appeared, they initially ignored them, explaining afterwards that they planned to deal with them at the end of their conversation.

Another area where there may have been a gap between our vision for passive assistants and how people understood it was in the architecture and the role of third-party apps. As part of our introduction to the system, everyone heard that features—including the most basic ones—were implemented by apps. Nonetheless, universally, participants spoke about the assistant as a whole. Their answers never treated different apps in our study as distinct or admitted the possibility of heterogeneity in app behaviors.

It is possible that this sort of metonymy was an artifact of our study, since we framed it as a test of the assistant in general (not any specific apps). However, researchers have observed similar confusion with existing devices and their third-party skills [138]. Our observations therefore portend that passive assistants (if their ecosystems end up following our prognosis) may face the same problem.

Detecting inappropriate requests We found that our permissions system worked fairly well for preventing data capture by the "misbehaving" apps in our study (§7.3). Participants denied a large majority of permission requests from these apps, whereas they allowed through most requests from other apps. Many also commented about the misbehaving apps during the interview phase after the interactive session, providing evidence that they were paying attention and that the misbehaving apps were anomalous and memorable.

> *"One was kind of strange. I think he said something like 'hotter weather' and there's something about like celebrity gossip, which is very strange."* (P23A)
>
> *"I kind of found it very spooky when they said 'bedtime stories' [the misbehaving app]. That made me very alert: why did they talk about bedtime stories right now? It's got nothing to do with what we were talking about, doesn't it?."* (P20A)

Some participants (less than a quarter of all cases) did say "Allow" to permission requests from misbehaving apps. This was primarily due to not paying attention or due to some amount of habituation (having gotten used to allowing all requests). Once, however, the participant allowed the request because they were curious about the promised functionality and did not perceive a risk in the information being shared with the app.

While some described the inappropriate permission requests as weird or even "spooky," most were not concerned by the misbehavior they observed, chalking it up to misunderstandings. Because users of voice assistants (and most other software) are so used to bugs and glitches, they saw this behavior as in line with those, rather than evidence of some sort of malevolent attempt at data capture. Some explicitly compared the misbehavior with difficulties they experienced using current voice interfaces, for example due to speaking English with an accent.

> *"Maybe it misinterpreted my words."* (P2B)
>
> *"It's kind of like when Siri gets stuff wrong."* (P10A)
>
> *"Probably because I have an accent, a really marked one. So actually when you were talking about the program coming up with the light bulb or something, probably it came from me because sometimes my accent makes me say the things or certain words with a different tone or something. And the program could misunderstand those types of things."* (P4B)

Consistent with our observation above, that participants did not clearly distinguish the assistant platform from the apps running on it, those who commented on the inappropriate requests did not clearly attribute them to apps, speaking rather about the assistant's mistakes.

How did people feel about runtime permissions and their streamlined versions?

One of the main research goals of this study was to collect first-hand feedback on the user experience of runtime permission requests.

Ask-every-time is annoying In the first interactive session, the assistant asked for permission on every potential data access. Largely as expected, everyone agreed that this resulted in too many permission requests. Many described this experience as "annoying" and expressed a strong desire for fewer interruptions.

> *"That's going to get on people's nerves, okay?"* (P3B)

> *"The first one was awful because it was, like, I have your permission for everything."* (P1A)

Because they resulted in significantly fewer permission requests, the streamlined permission modes (*Rules* and *Learning*) were received much more positively. However, beyond that, there was not much consensus about the two modes and their distinctive properties.

Advantages of *Learning* Between the two permission modes, a slight majority preferred *Learning*. Those who did expressed trust in the automation to accurately learn their preferences and explained that they were not concerned about it making mistakes and granting inappropriate permissions.

> *"Well I don't see any damage that it can do since it's not giving out any demands or orders anywhere. So I don't see much damage."* (P13B)
>
> *"Obviously not everything is perfect, right?"* (P22B)

Weaknesses of *Rules* Another reason people cited for preferring *Learning* was the cognitive overhead of the four permission choices in the *Rules* variant. The extra choices added text, increasing the time needed to read them, but also made the decision about which option to pick more complicated, since users had to think about whether they wanted to allow an app always or just once. While deliberation can be beneficial in that it reduces the influence of heuristics and cognitive biases [20], too much of it may make users unwilling to use the product.

> *"Less choices makes for a faster conversation."* (P13A)
>
> *"I had to really pay attention [...] I'm sorry, it was too much work, to be honest. It's just, I want it to be easy."* (P16A)
>
> *"I also think it creates a sense of paralysis by analysis."* (P17A)
>
> *"It was just a more intuitive experience. It was it was easier and you didn't have to constantly think if you wanted to allow or deny stuff because they are already knew."* (P21B)

Furthermore, the exact behavior of the rules in this variant was unclear, and nearly half of participants expressed some sort of confusion related to it.

> *"Little confusing on what to select? Because there's four options."* (P12B)
>
> *"I'm still a bit confused. [...] It kind of got me more distracted, because I'm having to stop to think about that."* (P14A)

Specifically, users were uncertain about the exact behavior of *Rules*: whether "always allow" referred to the specific app being always allowed, or if it was the specific speech they uttered (for example, any app could always access the location they just mentioned).

> *"Yeah, okay, so okay, it's not about the content, it's about the app."* (P17B)
>
> *"But that's the part that confused me because, well, music is on certain apps. So if I allow those apps then is it that I don't need to allow the music or is that allowing the music allows all of the music apps?"* (P5B)

Another issue that people brought up with the *Rules* variant was that rules were active forever: an app would retain its access unless explicitly changed through the review mode. Some assumed that was not the case, while others felt that it should not be the case. (This is in line with a desire for limited data retention found in Chapter 4 and other studies [110, 21].)

> *"Honestly, I assume, just as a regular consumer, I assume it was good for just that day and then it would probably reset again."* (P16A)
>
> *"I hesitate to do it once or because I might change next time. I'm not sure if next time I go I might change, so I debate on should I use always or should I just use it once?"* (P12B)
>
> *"I feel like if I'm allowing it once, then I won't regret the decision because I can always see that later. But if I allow it always and then it bothers me, I would have to go back and fix that."* (P14A)

Advantages of *Rules* Those who preferred the *Rules* variant expressed a desire for greater control of the assistant's behavior.

> *"I think that, for the sake of being more in control and knowing that once I say it is going to go away and it won't keep popping up."* (P10A)
>
> *"Sounds really like therapist stuff, but I feel like I have more support with [Rules mode]. I felt like there was more hand-holding going on. I felt like I had guidance."* (P16A)

This variant was also popular among those who distrusted the assistant's automation—or simply did not see it as beneficial—and did not want it making decisions on their behalf, especially ones that might have undesirable consequences.

> *"It's like the AI would be the one controlling it. And I think, in that situation, it's, like, why are you asking permission if you're going to not ask for permission later?"* (P23A)
>
> *"I don't think [Learning] that's a smart idea. [...] Humans are indecisive. One day, I might say I want this to be allowed, and the next day, I'm so annoyed because, why didn't you ask me?"* (P19A)

Non-use of the review feature The review mode (in either condition) also received mixed feedback. Only a minority invoked it during the sessions, mostly out of curiosity. Many said afterwards that they forgot about it, but some critiqued its user experience or even the need for it.

> *"I find it difficult to use that feature, actually."* (P22B)
>
> *"I don't need that. I trust her."* (P1A)

Participants were not universally opposed to the idea of a review feature, and many claimed they would use it, though with varying frequency. The most common reason people suggested for using it was if something suspicious happened. (As we saw in Chapter 4, this is also how the feature is used in current smart speakers.) Thus, the review feature's relative unpopularity may an artifact of our study, and it may prove to be more in demand with more prolonged use of the assistant.

For most, however, the review feature was about controlling the impacts of the apps, and making sure the device understood them, rather than an audit mechanism to verify that the apps and the automation were not behaving badly. This is consistent with how many participants used the permissions to select which suggestions they found useful, as opposed to rejecting only inappropriate requests.

> *"At the end of our trip you of course want to look over all your decisions and your planning. Just like [...] finalize everything."* (P14B)
>
> *"I wanted to see if not only I could see what apps I've approved, but also what I asked them to do. [...] So that, in that case, I wouldn't have any duplicate actions or events."* (P4A)

Did people trust the permissions?

Continuing the investigation of comfort and acceptability from Chapter 6, we wanted to know whether the permissions they trialed helped people trust always-listening devices more. We found that a number of participants, especially those who were less concerned about their privacy, did not see a strong reason for them.

> *"I guess in terms of the permissions that I see popping up [...] I see that more of like a redundancy. [...] Because I'm already asking it for these things by buying it. Buying it and having it in my house is almost like implicit consent as it is. [...] I don't see it as making me any more comfortable, honestly."* (P17B)

Instead, the prevailing sentiment was that they were interrupting, leading to unnatural conversation flows.

> *"You ask us if we want to allow or deny something that we said a while back, and we have to stop what we were talking about, so we can say whether we allow or deny what we had previously talked about."* (P9A)

Nonetheless, when prompted, a little under half of participants commented that permissions enabled their trust in the assistant.

> *"It would provide some comfort in knowing that I will know exactly what it needs to operate."* (P18A)
>
> *"I feel like they help me trust it more. [...] It makes me feel like I have the control for what I am allowing and I'm not allowing. So that gives me a sense of trust. Just because I feel like I'm the one making the decision."* (P14A)
>
> *"The fact that I did have to grant the permission, I feel, did make it a bit easier to ease into it."* (P3A)

Supporters of the permissions spoke about how they provided a greater degree of control, which they wanted. The fact that this preference was common but not universal could be a reflection of differences in the preferred level of control displayed by different people: while some people are interested in decision automation, others want only analysis automation and to make decisions themselves [165].

> *"Because it's not taking everything I say as a whole and storing it in some black hole where it can be hacked and given to somebody else. It's asking me what I want to store and whether I want to search that particular thing and I have an option of saying yes or no. There's some semblance of control."* (P20A)
>
> *"It just gives you a little bit more ability to keep things private in your mind. [...] If it's hearing everything, you know that it's already not private, but you're also wondering where this is going to. So that gives you a little bit more room to control it."* (P10A)

Many, including those who liked having the permissions, saw them as a way to control the suggestions (rather than a privacy feature).

> *"So for for me, the only time I would deny is if it was trying to help me too much. If it was something that I didn't want to do just yet."* (P4A)

Unfortunately, no one saw them as sufficient, even those who liked them. When presented with a scenario in which they were reading a credit card number out loud in the presence of the assistant, only one person stated that the permission system on its own would provide adequate protection; the rest explained that they would not feel comfortable relying on it alone.

> *"One of the main things that I think of is the app malfunctioning. What if the information did get through even despite despite the permission?"* (P15A)

Instead, they described other protective behaviors they would engage in, such as leaving the room with the smart speaker or unplugging the device.

> *"I would go to another room. I don't trust the microphones. I've been told that microphones are never off. Even when the phone is off, the microphone is never truly off."* (P16A)
>
> *"Better safe than sorry, like why not cover your ass? Wouldn't be that hard to turn it off. It would just be an added step to make me feel more comfortable."* (P14A)

Some pointed out that they were worried not only about the apps but also about the device itself compromising their privacy. This is an important reminder that the threat model we adopted is not fully aligned with that of real users.

> *"That doesn't have to do with the apps. All this has to do with Alva."* (P19A)

In addition to the less-interrupting permission modes that we tested (§7.2), we also surveyed our subjects about a design we refer to as "auditing," in which an app's permissions requests are approved automatically, but can be reviewed at any time, using the same interface that was provided for the other conditions. When we described this design to our participants, many thought it was preferable to all of the approaches they experienced first-hand.

> *"I feel like this one would be less overwhelming with the pop-ups coming up in your face—you know with accept and deny, accept and deny—which could probably get tedious after a while, they have to keep doing that. So if you have this version, where it accepts everything and then you could go and customize it and it's not so overwhelming, then maybe that would be a bit better and not so tedious."* (P15A)
>
> *"For me, I'm perfectly okay with that. That'd be the easiest approach."* (P9B)
>
> *"I'm kind of a lazy individual. I mean, I still get to control at the end, that's all that matters."* (P14B)

However, some had reservations about this approach, explaining that they felt that it took too much control out of their hands, and that it could be abused by apps.

> *"I will not be comfortable with that."* (P13B)
>
> *"I would like to have a preference, at least the first time, to choose review."* (P20B)
>
> *"I wouldn't. I always want to know, because the companies sneak in those random ones, that you don't have to approve to use it and it's not gonna affect usability, and they're just looking for some free data for their pockets. I like to catch that."* (P21A)

What other privacy controls and protections do people want?

Participants discussed a variety of other controls and privacy demands. The ability to turn off the device's microphone was considered very important, and helped our interviewees feel more comfortable with the device. However, studies of current smart speakers suggest that the mute button, present in all of them, is rarely used [122].

> *"There's a mute option, which is a big thing."* (P20B)
>
> *"Just having a simple on/off switch, or just saying verbally, 'Alva, turn yourself off!' "* (P21B)
>
> *"I don't want there to be a physical button to turn it off—I want it to stay on—just, I don't want it to pick up what I'm talking about at the moment because of how sensitive it is."* (P18A)

Some wanted not listening to be the default behavior and to only turn on always-listening mode for specific conversations—effectively, an on-demand always-listening mode. User studies have discovered analogous demands from users of existing smart speakers; for example, participants in Lau et al.'s study [122] said they wanted to see an incognito mode, including potentially a time-limited one.

> *"Maybe there should be a feature where it doesn't listen to you all the time, it's an option when you want to start a conversation."* (P6B)

However, five different people admitted that, if this were the case, they would forget to turn the always-listening mode off.

> *"I mean, if it would become part of my life, I probably won't remember. Every time somebody comes over to say, Oh, in this conversation, we have a computer listening to us."* (P8A)
>
> *"The logical thing to do would be to turn it off, but if they're always there, I think I would just forget that."* (P23A)

More than half of participant pairs independently requested a voice identification feature, in which the device should only respond to recognized voices and potentially treat different people or voices differently. This feature is familiar to many, as it is available in existing voice assistants (Alexa's Voice Profiles [13] and Google Assistant's Voice Match [92]). Voice authentication is now offered by many banks and has long been depicted in popular culture [169].

> *"You can program it to follow instructions from certain voices."* (P13B)
>
> *"You can select that Alva should only detect some voices. Maybe it's my voice. It can only do tasks after it hears my voice. And if it's someone else's voice, it just mutes."* (P11A)
>
> *"You have different options for privacy, and if there is any great variance on how it operates in one mode or another, then maybe it can recognize by voice preference or user preference."* (P21A)
>
> *"If it's a stranger in my house, don't be responding to the stranger. [...] Let's say someone breaks into your house and just start asking Alva information. She starts, yeah, 'they're gonna be at dinner at 7.' 'Oh yeah, I can come back and rob them at 8 PM because they'll be well into their dinner!' Or if you have a crazy stalker ex who somehow finds the key and breaks into your house or something like that. But yeah, I think that it should have some type of voice profiles that it recognizes."* (P19A)

Many also independently suggested parental controls as an important feature. While such controls are used relatively infrequently by parents of teenagers [88], the participants in our study generally sought protection for much younger children [161].

> *"Does Alva have a way to block off a toddler? Because our son can talk now. If he figures this out, he can send reminders non-stop every day."* (P7B)

Parents had different views about how much access their children should have. Some felt that the device should ignore children's voices altogether.

> *"It would be me and my wife and then the kids would be excluded."* (P5A)

Others simply wanted the assistant's answers and suggestions restricted to age-appropriate content.

> *"If something was going to be kind of inappropriate or like 18+ type content, then a pop-up or a preference allowance or warning would come up."* (P21B)

Other controls people came up with included limits on the times of day when the device would operate.

> *"If I could maybe set up some times when Alva should be muted, then I think that would be good. Like if it could only hear me in the morning or in the evening and not apart from that."* (P22A)
>
> *"When you want to start a conversation, you can say, 'listen to my conversations from 8 AM to 12 noon or something.' "* (P6B)

Another recurring suggestion was per-user passwords that would restrict access to data on the device. Participants may have been inspired by a variety of current systems, such as websites that re-prompt for passwords before accessing sensitive settings, banks that require PINs before revealing information, or app stores that ask for authentication before installing apps. Most relevantly, Alexa already offers the option to set a 4-digit "voice code" which is then used to confirm purchases and prevent accidental orders [11]. However, research has found that this approach does not meet everyone's security needs, especially in higher-risk scenarios [171].

> *"Maybe you can program it to have a password so that other people cannot be able to view whatever comes on the screen."* (P13B)
>
> *"Maybe there could be an option of putting a password that could enable Alva to recognize yourself as the owner that's using it."* (P2A)

Other suggestions included ''stop" words that would direct the device to stop recording, blocklists of specific words, and data sanitization if the conversation turns to certain topics. These approaches, while not available in present devices, appear imminently practical based on techniques in published research [202].

> *"If there's a certain word I'm saying, there will be a three-second delay and she just won't hear it."* (P1A)
>
> *"I would have a list of banned words. Financial, order, whatever. Social Security, tax, financial, money, cash. You know, things like that."* (P1A)
>
> *"A masking tool somehow…I would want some type of masking to automatically happen, if it's possible."* (P19A)
>
> *"When you speak that number it should really be able to detect that format and probably ask whether it needs to be deleted."* (P20B)

Participants brought up other privacy expectations they had for always-listening platforms. One was a rigorous review process that all apps for the device would have to undergo. This is analogous to the review process that apps must undergo before being published on smartphone app stores. Participant 17B explicitly drew this comparison, explaining that the rigorous vetting process in the Apple App Store won their trust. Today's voice assistant platforms already require third-party skills to undergo "certification" [12]; however, this verification process may become more difficult for passive listening assistants, if they allow their apps the same level of freedom and flexibility allowed by our architecture. We will explore some possibilities for this process in Chapter 8.

> *"The main security feature is I would want Alva to monitor anything that looks suspicious. Let's say, for example, there's an app that is supposed to do restaurant reviews, but it turns on the camera and turns on the microphone and sends it somewhere else the whole time, that's a third party, for marketing data or something. I would want Alva to notice that and let me know."* (P17B)
>
> *"When an app comes on your Play Store, is it cross-checked to see that it's not something that can cause damage? Does it go through any verification?"* (P20B)

Multiple participants said that they wanted to be compensated in the event a data breach occurred. Some responses suggested a belief that there are existing policies or laws that provide for this. Such misunderstandings of privacy regulations are long-standing and well-documented [208].

> *"You get your money back and like a compensation type of thing. You know, like in the privacy article."* (P15A)

> *"If something does happen, I can contact somebody, I'm not really screwed. Like let's say something happens and I need to contact somebody at the Alva company to fix it. I'd like somebody to be there that I can contact that would be able to help me out, as opposed to, like, if something goes down with your Gmail account or whatever, who are you going to call?"* (P17B)

Data minimization was another approach that one respondent hoped to see adopted. requirement of GDPR One respondent explained that they hoped developers would only collect the data they need, a strategy clearly recognizable as data minimization, which is a requirement of regulations such as GDPR [71].

> *"If it's not using it to work or to search for us, then it doesn't need it and it shouldn't sell it. I wouldn't want to sell anything beyond usability, functions, things they have to."* (P21A)

Participants also discussed other privacy factors that they found important. Among them was having a privacy policy that promised to respect their data, as well as providing security disclosures. These may be satisfied by requirements that arise from laws such as CCPA [39].

> *"I just want an assurance of my privacy and maybe its safety and reliability information."* (P2B)

Others brought up that their decision about adopting the device would be influenced by the manufacturer's reputation, as well as their business model.

> *"Who is the manufacturer? It depends upon that too."* (P3B)
>
> *"Who exactly is the official producer of this device? Is it some well-known company or is it just more of a kind of start-up kind of device?"* (P23A)
>
> *"I would be concerned about the company collecting and selling data, so I would probably search about how they operate."* (P21A)

7.5 Discussion

This study collected people's perceptions of passive listening, their privacy preferences for it, their reactions to runtime permissions, and their suggestions for other privacy controls. In this section, we reflect on the main themes and lessons.

Many will welcome passive listening One basic observation from our study is that there was no wholesale rejection of passive listening as creepy or excessive. Of course, our participant sample is biased, since we recruited people who were willing to be interviewed and have their interviews recorded. Many were already owners of smart speakers and other IoT devices. Still, we believe that smart speakers have paved the way for passive listening: our interviewees described it as a natural extension of present-day functionality. Even if our sample is not representative, there is clearly a market opportunity, likely from a sizeable fraction of the population.

Concerns center on actions and consequences As we have emphasized, openness to passive listening is not equivalent to not caring about privacy. In fact, privacy concerns are pervasive and were raised, in some shape, by nearly everyone in our study.

Promisingly, the concerns mentioned most often seem plausible to overcome. When it comes to passive listening, people seem most concerned about impactful activities: an assistant taking autonomous actions that end up having financial, social, or personal consequences for the user. From a designer's perspective, this appears straightforward to address by ensuring the assistant (or app) *confirms* with or *notifies* the user about any actions it is taking, such as making purchases or setting alarms. Ideally, this feedback can be provided over multiple modalities, in case, for example, the user is too far to see the display, or, conversely, the environment is too loud for the assistant to be heard.

Though these outcomes are top-of-mind for our interviewees, and would certainly be unwelcome, we observe that a mistaken purchase or incorrectly set alarm may arguably not be a privacy issue. However, other examples, like misdirected messages or unintentional invitations, are more clearly violations of a user's contextual integrity.

Standard sensitive content should be excluded When it comes to the assistant simply hearing information (as opposed to taking actions), the concerns voiced by participants—at least, ones they came up with over the course of our interviews—were not particularly heterogeneous. They centered primarily around a few sensitive data types, such as financial information or gossip.

An implication of this finding for system developers is that they can assuage users' concerns, to a high degree, by implementing data sanitization, filtering, or other deny-listing approaches that we discussed in Chapter 5. While these will vary in how easy they are to implement (detecting credit card numbers seems much more tractable compared with identifying gossip), this appears to be a promising research direction and likely an effective way of winning the trust of many potential users.

Despite the emphasis in our interviews on sensitive data types, the level of care participants exhibited about privacy suggests they would not be okay with indiscriminate data collection—even the deny-listing approaches may not be sufficient—thus justifying our

search for a more protective permission system.

Intra-household controls needed While not a focus of this study, our interviews provided evidence for the well-known fact that people are concerned about protecting their privacy not just from apps, strangers, and other third parties, but also within the household [228, 86, 7, 25, 115]. One approach that many suggested was voice identification, accompanied with access controls that would limit one's data to just themselves. Other ways of enabling intra-household privacy could be the subject for more research.

Permissons, with architecture, help catch bad requests Our testing illuminated both positive and negative aspects of runtime permissions for a passive assistant. The permissions seemed to show potential as a way of fending off inappropriate data access by apps, as most participants effectively identified and blocked the misbehaving apps in our study. For many, permissions also increased their trust in the device and gave them a sense of control, which they described as very important, especially for a device in such a sensitive setting.

Proposed permission designs show promise, face adoption challenges As a user experience for voice assistants, runtime permissions showed some promise, as participants understood them and were able to use them effectively during the interactive sessions. The designs were also quite successful methodologically, producing many helpful insights about potential choices for permission systems. However, we caution that none of the permission modes we tested is likely to yield a user experience that would be acceptable for a real product. As predicted, no one—even those who were more privacy-conscious and wanted more control–was happy being prompted every time an app wanted to access data. Reactions to the less-interrupting designs we trialed were much more positive, as participants appreciated their streamlined nature, but they exhibited limitations of their own.

The ***Rules* design** provided the added options to "always" allow or deny requests for specific combinations of apps and data types. It offered people, especially those who were less trusting of the system, a greater degree of control. That sense of control may be misleading, however, since the relatively permanent nature of the rules may lead people to forget about the permissions they granted. This is exacerbated by the fact that many were confused about what exactly they were allowing. Finally, a majority felt that having four options on every request was too cognitively taxing. All together, these pain points suggest that the *Rules* design, in its current form, would face challenges if adopted as a general-purpose permissions approach.

In contrast, the ***Learning*** design has the advantage of a much simpler user experience. One limitation—the dual of the observations above—is that a sizeable minority of participants (even in our, potentially biased, sample) seemed unwilling to give up control over data access to a black-box algorithm. The development of an algorithm that can ef-

fectively learn people's preferences across a variety of contexts also remains an open, and daunting, research question.

One interesting challenge for using machine learning on participants' preferences is the way people used permission requests: they denied them when they considered the service unnecessary. A system that tries to use this as a training data point in a model may come away with the wrong conclusion: that the data access request was inappropriate, when in reality it was just this instance that was not useful. Ideally, a model would be able to capture variations in people's preferences such as these, or other contextual factors, such as time. Indeed, this should be a fruitful avenue for further research. But, for now, these challenges cast the practicality of the *Learning* approach in further doubt.

One more approach that we surveyed our subjects about was "**auditing**," in which permissions were approved automatically, but subject to review after the fact. For the more privacy-conscious, this was unacceptable, but the majority actually preferred it, since it did away entirely with the universally deplored interruptions from the permission requests. Yet our findings suggest that adopting this variant would likely lead to poor privacy outcomes. People would be unlikely to make use of the review feature, as evidenced by this study, the results of Chapter 4, and experience with other systems. And this would be exacerbated by the misunderstanding many users have about the distinction between the assistant itself and third-party apps for it.

Design proposal How should we move forward? More research is needed to explore whether there are other permissions designs or approaches, that we did not test here, which would yield a more favorable user experience and equivalent (or better) privacy protections.

What if we had to build a passively listening voice assistant today? Until better approaches are available, the most effective tactic may be to combine the strategies that emerged as the most promising from this study. Concretely, we imagine that the assistant would have some of the following features:

- Any actions that trigger consequences beyond purely looking up information would be subject to manual approval and would receive multi-modal feedback (e.g., using voice and on the display).
- By default, the platform would identify, block access to, and immediately erase any financial information and other known sensitive topics. (Perhaps users could review a list of such topics when they set up the device.)
- Users would need to opt in to "online" passive listening for specific conversations or short periods of time. When opted in, permission requests would be automatically approved, though still auditable after the conversation has ended.

- During these passive sessions, users should be made aware of which apps are accessing their conversation.
- The device would feature voice identification (to restrict users' access to their own data) and parental controls.

While this design may not achieve perfect privacy, it would significantly enhance it compared with other approaches where apps might always be listening, and it would address many of the concerns and user experience pain points that we discovered as part of this study.

Chapter 8

Evaluating Transparency Mechanisms for an Always-Listening Assistant

We studied whether people could detect if an always-listening app is malicious by having them submit sample speech and providing instant feedback about whether or not the app considered it relevant. Success was high for more obvious attacks, like those targeting financial information, but lower for more subtle ones, and people working in groups were generally more successful.

8.1 Introduction

This chapter studies the ability of humans to detect whether services for an intelligent voice assistant are trying to violate its users' privacy.

In our previous studies, we saw that install-time permission requests may be ineffective (Chapter 6) but that runtime permissions are annoying (Chapter 7). One approach that received positive feedback in the latter study was auditing. In this permission design, permission requests from apps would be automatically approved by the platform, but users would be able to review them after the fact. While some of our interviewees felt that this design relinquished too much control, others found it sufficient and appreciated the lack of interruptions it enabled.

Auditing is one example of a broader approach we refer to as *transparency*. Under this regime, users do not exercise direct control over what the app does. Instead, the architecture allows the exploration and analysis of an apps behavior—before or after it takes place—so as to enable a determination about whether it behaves appropriately or not. This decision might be done by end-users or workers specially hired and trained by the

voice assistant platform. (See additional discussion in §5.8.)

A natural concern about the transparency approach is regarding its efficacy. Would it enable users to catch malicious or misbehaving apps? Or would privacy violations remain undetected? This chapter is dedicated to studying these questions.

8.2 Approach

To address our goal of identifying malicious always-listening apps, we propose an evaluation mechanism for human users and an architecture that enables it. In this architecture, apps get to decide what is relevant to them, but are subject to black-box testing, which enables humans to verify if the apps are accurate in their assessment. In this section, we motivate and describe our approach to studying this problem.

Black-box testing Black-box testing is a commonly used technique for exploring the behavior of software when its internals are opaque. The software is run in a constrained execution environment and its effects are observed. This is common, for example, when analyzing malware samples [181] or studying privacy behaviors of apps in smartphone app stores [178]. Our approach applies the same technique to detecting malicious apps by examining their behavior, rather than the internals of any machine learning model. However, doing so requires some modest assumptions, which we outline below.

Assumptions In practice, most apps are unlikely to need access to *all* audio all of the time. Instead, they will probably distinguish conversations that are relevant to them from those that are not. While such a "relevance detector" would likely itself be a machine-learning model and thus a black box, we will assume that it is separable from the rest of the app's functionality and is accessible for security analysis.[1] This can be mandated by the voice assistant platforms, for example, as a condition of inclusion in their app stores. With this architecture (which we will elaborate in §8.4), a tester could provide some audio or text and see if the app considers it relevant. Anything that is considered relevant, but is outside the purview of the app, will be cause for suspicion.

Limits of automation Even if we assume the existence of relevance detectors, it remains a challenge to detect malicious apps. This is because defining what is in scope for any given app is a difficult problem. Proactive apps may be created to support a wide variety of use cases, from remembering information to offering relevant advice on just about any subject that can come up in a conversation [200, 215]. The breadth of potential app scopes, coupled with the ambiguity inherent in many natural-language conversations, means that any simple heuristics are likely to be insufficient. For example, we could assume that no app should be hearing financial or health-related topics, but this may preclude a reminder

[1] A non-security reason for why an architecture with a relevance detector may be practical is that services may not want to stream *all* audio to the cloud—just the functionally relevant portions.

app from being able to help users with legitimate use cases, such as scheduling reminders for medical appointments or bills that are due.

Furthermore, privacy theories and research have shown that an app that behaves counter to user expectations—even if it does not capture specifically sensitive information—can make people feel that their privacy has been violated [160]. If we want to avoid privacy violations, we therefore need to be able to determine precisely what any given app should or should not be hearing. Unfortunately, coming up with an unambiguous policy about this is difficult even when communicating with other humans, let alone algorithms. Thus, until dramatic advancements in the capability and reliability of natural-language processing occur, the only way to resolve at least some of these ambiguities will be for humans to exercise their judgment.

Humans in the loop There are several places for humans to enter the malicious-app detection process. Analogously to the ecosystems of modern smartphone apps, the app platforms for IVAs may employ workers to review apps, independent groups or organizations may step in to provide judgment, or users themselves may wish to evaluate apps before installing them. Even if machine learning is utilized to detect malicious apps, humans will likely be called upon to generate ground truth: training examples of what a particular app should hear—and what it should not.

Research question If we need to rely on humans as part of the critical objective of detecting malicious apps, it is important to understand how well humans actually perform the expected tasks. If people prove to be especially effective, then developers can go ahead with systems that rely, to a greater extent, on humans in the loop. On the other hand, if people struggle, then platforms will need to rely on software more than humans, dedicate more resources to training and educating users, and research additional ways of minimizing human involvement. Thus, our research question is to understand **how effective are humans at detecting malicious apps?**

8.3 Research goals and contributions

To address this research question, we developed the concept of a *Test Drive* for an always-listening app. Following the notion of black-box testing, a user performs a Test Drive by supplying samples of conversations (or fragments thereof) and learning whether or not the app would consider them relevant. People can utilize Test Drives to identify suspicious app behavior by supplying off-topic (or otherwise inappropriate) examples and observing whether or not they would be accessed.

We propose using Test Drives as a way of measuring how well people detect malicious apps. To enable this, we add another key requirement for Test Drives: that they provide instantaneous feedback about whether the app would have heard the input or not. This

way, the user can make an immediate decision about whether the app is benign, malicious, or merits further testing. Interactivity has the added benefit that if a user observes potentially concerning behavior—but not yet clear evidence of malevolence—they can devise more targeted speech examples to investigate the behavior.

In our study, we contribute to the field of human-computer interaction by investigating Test Drives as a technique and using them to shed light on people's ability to discover malicious apps. More specifically, we:

1. **Introduce a voice assistant architecture that enables auditing**
    - As well as a specific auditing technique: Test Drives
2. **Understand how people make use of Test Drives**
    - What kinds of inputs do people provide and behaviors do they test for?
3. **Evaluate Test Drives as a technique for detecting attacks**
    - Are people able to detect attacks?
    - How does task formulation affect people's ability to detect attacks?
4. **Explore Test Drives in other settings**
    - With groups working collaboratively
    - Without interactive feedback

8.4 Methods

The goal of our research is to understand how good people are at identifying malicious always-listening services, which we accomplish through the use of Test Drives. In this section, we describe our assumptions, including the voice assistant architecture and threat model, then detail the design of Test Drives and our experiments.

Threat model

We begin by assuming that, like today's IVAs, future passive listening assistants consist of the platform—the assistant's operating system and first-party features—as well as a wider ecosystem of third-party apps.

The threat agent we are protecting against is a developer of a third-party passive listening app, who wishes to obtain conversations from the user, and information therein, that are outside the scope of the description of the app, as provided by the developer. In other words, apps come with a description of their purpose and behavior; any attempt to access information irrelevant to those is considered an attack.

Attackers may have different motivations, but we assume they will most often be interested in financial gains and will therefore seek to obtain information that can be sold to advertisers or on illegal markets. In fact, advertising libraries, embedded in apps, may be the source of the attacks, as happens currently with smartphones [164]. Attacks may also vary in sophistication. A naive app might attempt to listen to everything. More sophisticated attackers may go after specific information, such as credit card numbers. They may also do this *in addition* to providing their officially stated functionality, which can act as a decoy.

Under our threat model, the platform is trusted with continuous recording of users. We will also assume that the platform transcribes the conversations into text, precluding apps from directly knowing users' gender, age, emotions, and other prosody and vocal characteristics. However, the content of the speech may allow apps to infer some of these characteristics; we consider these and other inferences out of scope.

Architecture

To enable interactive evaluation of passive listening applications, we propose and require a specific architecture that IVA platforms must enforce.

App developers participating in the ecosystem architect their apps in two parts: a "relevance detector" and the remainder (core) of the app. A relevance detector takes as input a conversation and outputs a binary decision: whether or not the conversation is relevant to the app. It runs in a *stateless, sandboxed* environment and has no access to any other inputs. App developers supply the platform with an executable version of their relevance detector, for example as a pre-compiled package or binary file. If the developers wish to make an update, they must provide an updated binary. The same version is used both for

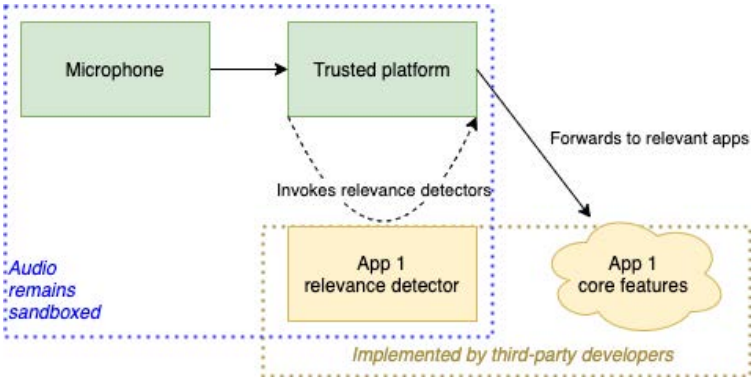


Figure 8.1: **Test Drive architecture**: overview of an intelligent voice assistant architecture that enables support for Test Drives

testing and evaluation, as well as when the assistant is actually running.

This architecture ensures that developers are not able to silently issue updates that introduce unverified behaviors. It also prevents them from including a switch that makes the app behave differently after being installed. Furthermore, it enables forensic investigation, as conversations can be "replayed" to see the relevance detector's determination.

The platform runs relevance detectors from each app simultaneously in a sandbox. If an app's relevance detector finds a conversation relevant, that data is transferred by the platform to the core app. (Figure 8.1 provides a visual overview of this flow.) The core of the app processes conversations that have been classified as relevant and acts on them to implement the app's functionality. It is able to communicate over the Internet and may run on third-party servers.

What an app does with data that *is relevant* to the app is not covered by our threat model: apps are assumed to use it without limitations. Instead, the goal of our architecture is to ensure that only content that is truly relevant to an app passes the relevance detector and is transferred to the app core. Our system enforces this through a novel transparency mechanism: Test Drives.

Test Drive design

The core idea of a Test Drive is for a person to supply some input and see whether an always-listening service would hear it. This allows them to evaluate whether or not the service may inappropriately record the given conversation. A Test Drive's role in the app lifecycle is an open question, leaving multiple possibilities for *who* the people evaluating the app should be: individual end-users or dedicated testers. We imagine that, similar to today's open-source applications, some experts or organizations will engage in more thorough review, while the majority of users will rely on reputation. However, the Test Drive interface makes the barrier to entry much lower—there is no need for programming knowledge—allowing for end-users to test apps as well, before deciding to install them.

The exact user experience of a Test Drive is subject to further research, also allowing for variations. For example, to enable an evaluation that is larger-scale and more ecologically valid, a real-world deployment might allow users to supply, as an input to a Test Drive, the last day of conversations captured by their device.

We note that there is a spectrum of attacks a malicious application could carry out, and some may be stealthier and more rare, for example due to having a very narrow set of triggers (e.g., a specific combination of words that initiates an attack) or being designed to happen probabilistically. Test Drives are one mitigation strategy among many and may not be an ideal for such rarer attacks; instead, they are designed to target behaviors that affect many users. However, Test Drives may be complemented by other techniques, such

as large-scale automated analysis and after-the-fact detection and verification, which are also enabled by its architecture (as outlined above).

In designing the Test Drive experience, we worked to balance realism with a level of consistency and reproducibility necessary to enable between-subjects comparisons.

Text inputs

While voice assistants and skills are usually invoked by voice, we chose for inputs in our study to be provided using text (Figure 8.2). This matches our assumptions about the architecture of future always-listening assistants, where (like in present-day systems) the platform is responsible for text transcription, and the apps operate on already-transcribed text. This approach also has several practical advantages for our study. It eliminates ambiguity and misunderstanding that may come from speech transcription. It enables greater anonymity for our participants. And it allows a single tester to provide conversations where multiple people are speaking.[2] On the other hand, allowing audio inputs would have made for a more visceral experience and could potentially help users better imagine themselves saying the example speech to the device.

Minimum engagement requirements

In a production system, the motivation may come from a platform paying workers to evaluate apps or from users wanting to test whether an app they are about to install will harm them. In a laboratory setting, we cannot replicate these incentives. However, research has shown that participants in experiments who are asked to role-play their be-

[2]There is no rigid syntax for doing this, but our examples—seen by all participants in the study–denoted a conversation as follows:
Person 1: . . .
Person 2: . . .

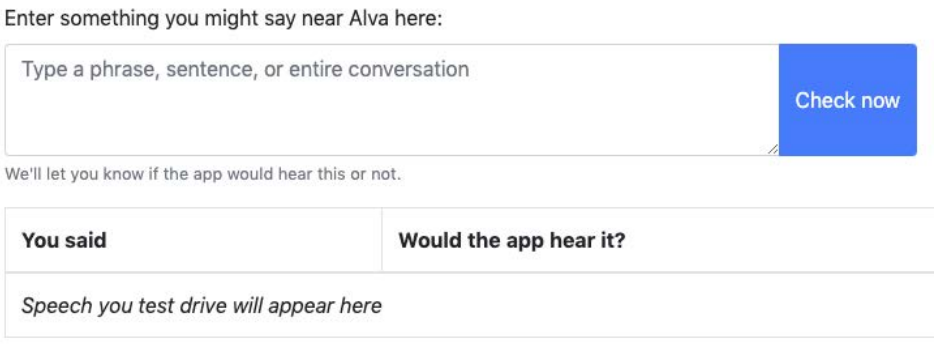


Figure 8.2: **Test Drive input field**

havior in certain security-sensitive situations make largely the same decisions as they would in real-life settings [113, 72]. Therefore, we asked participants to imagine that they were deciding whether or not to install the apps in our study for their own devices. Additionally, to ensure a baseline level of engagement, we required at least five utterances before participants could proceed from a Test Drive. In collaborative mode (described below), the minimum threshold of five utterances was applied to the entire group, rather than its individual constituents.

Wizard of Oz

Another major design decision we made was to implement our Test Drives using a Wizard of Oz method. This technique (introduced and utilized in Chapter 7) involves simulating the behavior of a system for the participant, rather than actually implementing the feature. In our case, this meant that, rather than training a machine learning model to classify whether or not some input was relevant to a service, a researcher read the input, made a decision, and supplied the output. The choice of this technique introduces a tradeoff between relying on the researcher's personal judgment, with some inherent degree of randomness that entails, and the behavior of a real machine learning model, which would have to be custom-trained for the novel always-listening tasks in our study, potentially producing too many false positives and false negatives that would overwhelm the intended behavior of the app (benign or malicious). In that situation, misbehavior identified by participants would more likely be a manifestation of errors in the model, rather than a true example of the service being malicious. While we opted for the simulation approach, future work may explore the efficacy of the fully automated methodology.

Implementation

We implemented our study using a custom web application. Participants completed our survey at their own pace. During the Test Drive stages, after the participant provided a sample input, it would be immediately sent over a WebSocket connection to a researcher. (The same researcher mimicked the machine learning model for all Test Drives in the study.) The researcher would—as quickly as possible—classify the content of the submitted utterance according to its relevance to the app's stated purpose and interest to a malicious app. (See §8.4 for more about the classification process.) Based on this determination, one of the following responses would then be shown to the participant:

- "The app would hear this."
- "The app would *not* hear this."
- "The app would hear *only part* of this."

Due to the architecture of our system, in most cases, participants would receive a response to their input in under ten seconds (median response time: 8.0 seconds). We did not explicitly check whether participants realized that the "AI" they were interacting with was actually a human, but their open-ended responses suggested that few, if any, realized the decisions were not being made by a machine.[3]

Choice of services

For the Test Drives, we needed to choose specific always-listening services for our participants to test. We made our selections with two goals in mind. First, the nature of the service had to justify why it might need to listen continuously, as opposed to being invoked by a wake-word. Second, the purpose of the app should be one that most people would plausibly consider useful. Based on these criteria, we used the following apps in our study:

1. **Reminders**: automatic reminders for scheduled or planned activities
2. **Weather**: answers to queries, plus automatic weather information for planned outings
3. **Cooking**: recipe and cooking advice

(Complete descriptions of the apps, as shown to participants, can be found in Table D.1 in the appendix.)

Since similar services are popular today, we hypothesized that their passively listening variants—which would enhance their convenience—would similarly be attractive to potential users.

All participants saw the same three apps in the same order.[4] However, the apps' behavior was not always the same: for every participant, two of the apps behaved maliciously.

Defining malicious apps

To test whether people can detect malicious apps, our apps had to sometimes act maliciously. After considering different threat actors and types of attack, we decided on the following four attack conditions.

1. **Financial**: any information related to money or finances, as well as credentials (usernames/passwords) for any service

[3] Participants in our pilot were similarly surprised to find this out.

[4] We chose to keep a consistent order, so that the only variable that was changed between participants was whether the app was malicious, rather than which apps the participant had seen previously.

2. **Sensitive**: any (other) information that people might consider private (subjects involving health, crimes, relationships, potentially embarrassing or compromising details)
3. **Personally Identifying Information (PII)**: names, birthdays, and addresses, or similar information that might uniquely identify an individual
4. **Overcapture**: *any* conversation or details that the app would otherwise consider irrelevant, if it precedes or follows text that *is* relevant to the app

While these attacks (1–3, in particular) are fairly straightforward, and could plausibly be detected by automated means with minimal human involvement, we chose them because we felt that they were most representative of attackers' motivations, and that they could provide valuable baseline information about people's ability to think adversarially, allowing future studies to focus on more sophisticated attacks.

When Test Driving an app designated as malicious, we considered the attack functionality to exist "on top of" the intended feature set. In other words, we applied these attacks as follows:

- If the submitted utterance was relevant to the app's purpose, we said the app would hear it.
- If the utterance was *not* relevant to the app, but contained information pertinent to the attack condition (e.g., a credit card number in the *Financial* condition), we said the app would hear it.
- If the utterance was not relevant to the app and it did not match the attack condition (even if it matched a different attack condition), we said the app would not hear it.

Study procedures

All participants in our study went through the same survey flow. (A complete survey instrument, including visuals, can be found in Appendix D.3).

1. Introductory questions and an explanation of always-listening apps and the concept of Test Drives
2. First Test Drive of an app, which—unknown to the participants—was malicious
3. A *treatment* page that revealed that the app had been malicious and provided additional information about malicious apps (see Figure 8.3 for example)
4. Two additional back-to-back Test Drives
5. Follow-up questions about the Test Drive experience

After each Test Drive, we asked the participant for their decision about the app: whether they would install it and/or whether they thought it was malicious. To collect confidence levels, we posed this question as a 5-point Likert-type scale, from "very likely malicious" to "very likely well-behaved." We then asked participants to explain their reasoning for this decision, and provide general opinions about the Test Drive interface and how they would use it, in separate open-ended questions.

The app in the initial Test Drive was always malicious. In the two post-treatment Test Drives, one app was benign and the other was malicious. The order of the benign and malicious apps was randomized, as was the type of attack. The attacks before and after the treatment were chosen independently (i.e., *with replacement*). The informational treatment consisted of an explanation of the type of attack the participant experienced during their initial Test Drive, along with one example of a conversation the malicious app would have inappropriately heard.

Survey variants

While the Test Drive experience, and the overall survey procedures, were the same for all participants, we introduced some minor variations into the study design. We did this because there were design choices that we considered equally valid, and we wanted to test their implications for people's performance.

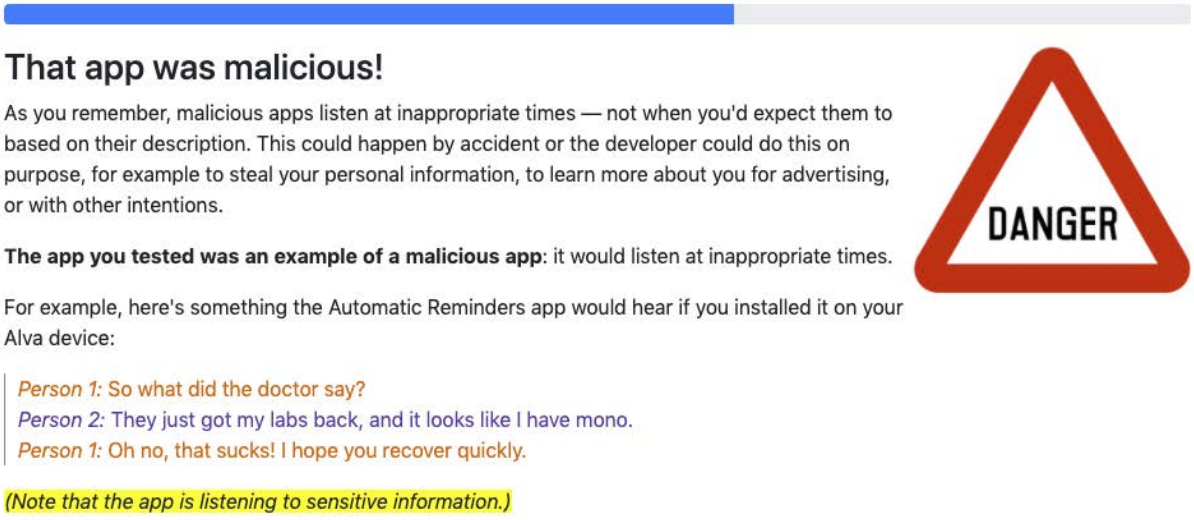


Figure 8.3: **Informational treatment**: how participants were informed that they had interacted with a malicious app (in one specific condition)

"Install" vs "Test"

The opportunity to Test Drive apps could be useful to end-users who are deciding about whether or not to install an app. We hypothesized that, as part of their evaluation process, users would check whether the app was listening for anything inappropriate. We further thought that this would provide insight into the concerns people have about always-listening devices, since participants would enter the type of things they would least want an assistant to hear.

> *Hypothesis 1:*
> People will test for inappropriate behavior as part of deciding whether to install an app.

We therefore formulated the task for our participants as deciding whether or not they would install the app they were Test Driving. We asked them about whether they thought the app was malicious only as an incidental question in the post-treatment Test Drives. We refer to these procedures as the *Install* variant.

During an initial round of data collection, we observed that participants were expending significant effort submitting *positive examples*: speech the app *should* be hearing. This makes sense from the perspective of a user deciding about whether to install an app, since they want to know if it actually works.

> *Hypothesis 2:*
> When deciding about installing an app, users are primarily concerned with its functionality.

Unfortunately, this meant that users were not fully dedicated to the task of identifying malicious apps, and therefore findings from the *Install* variant would not be the most effective measure of our primary research question, people's ability to detect malicious apps. We therefore introduced a second version of our procedures, the *Test* variant.

> *Corollary 2:*
> People will be more effective at detecting inappropriate behavior if explicitly asked to do so.

In the *Test* variant, we explicitly instructed our participants that their task was to identify whether the app they were Test Driving was malicious. We did not ask them whether they

would install it and told them: "It doesn't matter how well the app works, or whether you yourself would want to use it."[5]

Nudges and wording changes

Within the *Install* and *Test* variants, we trialed some minor wording differences to examine the difference they might have on people's performance.

> *Hypothesis 3:*
> Nudges may induce people to more effective evaluation.

Initially, in the *Install* variant, we chose to describe attack apps as "misbehaving" as part of our explanations, in order to have wording that was more neutral. However, after observing that participants were expending higher-than-expected effort testing the functionality of the apps (rather than their maliciousness), we started referring to these apps as "malicious," in case this wording helped draw attention to the threat potential users faced. (In both versions, the participants' primary task was still deciding whether they would install the app or not.)

In the *Test* variant, we retained the "malicious" wording but tried other nudges. In one version, we suggested participants try attacks other than the example that was shown to them during the treatment. In another, we reminded them, as part of the Test Drive instructions, about their answer to an earlier survey question in which they provided specific examples of speech they would not want a malicious app to hear.

Collaborative evaluation

Up to this point, we have assumed that a Test Drive involves an individual evaluating an app for misbehavior *on their own*. However, there is no reason why people should not be able to collaborate, especially as, in any real systems, detecting malicious apps is overwhelmingly likely to be a collective activity by the community at large (including the platforms and their employees, end-users, and other interested parties). We were therefore interested in seeing how collaborative evaluation would play out.

> *Hypothesis 4:*
> Groups of people can (also) detect malicious apps.

[5] An additional difference between variants was that participants in the *Test* variant were explicitly told that apps *could* be malicious prior to their first Test Drive. Participants in the *Install* variant received this information only *after* the first Test Drive. However, in both variants, participants were only given a specific example of a malicious app and informed that the app they had tested was, in fact, malicious *after* the informational treatment (i.e., the first Test Drive).

While we found it reasonable to assume that groups would out-perform individuals, we felt that it was also plausible that they may be subject to *informational cascades* [28], in which members would trust others' decisions rather than making up their own minds. We consider an in-depth investigation of groups, their pitfalls, and their comparison to individuals to be subjects for future work. As the first study utilizing Test Drives, we see our role primarily as exploratory.

Thus, we created another version of our study, where Test Drives were collaborative rather than individual. Participants in this collaborative mode followed the same overall procedures as those in individual mode, and their first (pre-treatment) Test Drive was done on their own (to get them more acquainted with the format and the task). However, in the two post-treatment Test Drives, participants saw not only the results for the utterances they submitted, but also utterances, and their classifications, from everyone in their group who had participated in the Test Drive so far. (Participants worked on the study simultaneously, and the results were synchronized between the subjects in real time.)

The other difference in collaborative mode was that participants could flag any heard utterance as evidence of "malicious behavior" by clicking on a button next to it (Figure 8.4). We introduced this feature to get a more precise measurement of which utterances arose people's suspicions, especially when they themselves had not submitted one. Participants in collaborative mode did *not* see others' decisions about whether the app they were testing was malicious, thus eliminating the chance of conformity bias, since everyone made up their mind independently.[6]

Non-interactive detection

One of the major assumptions and design decisions in our study is that users should have real-time feedback during their Test Drives about whether or not the app would hear the

[6]We further observed that reports of malicious behavior by one participant did not necessarily induce others to report the same utterance, nor did they guarantee that they would deem the app malicious.

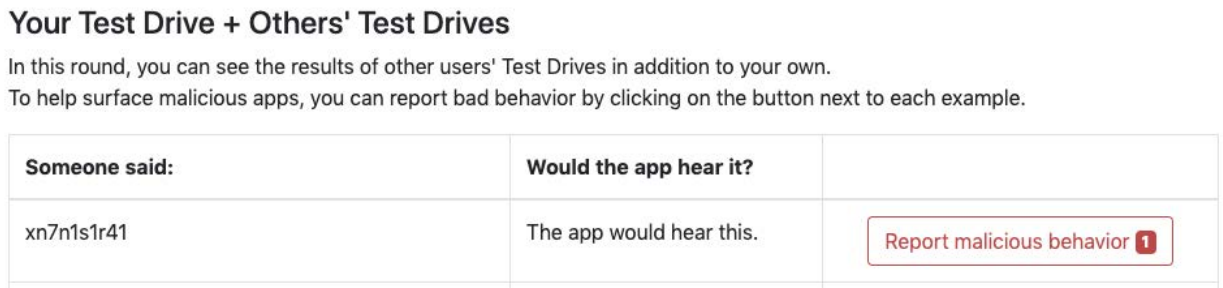


Figure 8.4: **Collaborative Test Drive interface**

provided utterance. We discuss the advantages of this above, but we also wondered how necessary this was.

> *Hypothesis 5:*
> Participants can also detect attacks without interactive feedback

To answer this question, as the final step in our study, we asked participants in both individual and collaborative mode to submit utterances that they "would use in a Test Drive to find out if the app is malicious"—but did not provide them with feedback about whether the app would have heard them or not. As part of our results, we will analyze how much of a difference this made.

Analysis

Each utterance submitted by our participants was classified on the fly, with the result immediately returned to the participant as described in §8.4. The classification was done along two axes: whether the utterance was relevant to the stated purpose of the app being tested (e.g., was it about cooking?) and whether it contained information relevant to *any* of the attack apps (e.g., whether it mentioned financial details, PII, etc.). The combination of these two decisions determined what was reported to the participant. Any topically relevant inputs would be reported as heard by the app, but so would irrelevant inputs if they contained information of interest to ("matching") the current attack app.

These ratings also serve as the basis of our analysis, as they allow us to characterize and quantify the types of attacks our participants were trying to uncover. For example, if a participant submitted the utterance "my password is hunter2" while Test Driving the cooking app, it would be classified as not relevant to the functionality, but of interest to the financial attack app; during analysis, it would contribute to the total count of financial attacks. There were no formal criteria about what constituted relevance to a specific app; the coder relied on their best judgment about whether a proposed utterance was one the given app would act on, based on either the app descriptions (Table D.1 in the Appendix) or the attacks as outlined in §8.4. Relevance to the app's purpose could be rated as partial, in the event that only a subset of the input was deemed relevant.

To determine whether participants correctly identified malicious apps, we used the Likert-type question that asked them if they thought the app was malicious (binarized as "yes" if they said "very likely" or "probably" and "no" otherwise). However, we also consider an alternate metric, attack discovery, in §8.5.

All ratings and classifications in our study (e.g., whether some input was relevant or not to an app) were done by a single coder, in real time during the Test Drives. This necessarily means that the ratings are less reliable than they would be if they were done by

multiple coders. We chose to have only a single coder during the Test Drives to ensure that participants would receive feedback on their inputs as quickly as possible. We did not re-code the utterances for analysis so that our data remained consistent with what the participants saw. However, to better understand the potential variance in classification judgments, a second researcher re-coded a subset of the utterances. Based on a power analysis of the total number of utterances (2,897), the number of utterances re-coded was 350. We then measured the inter-rater reliability, yielding Cohen's $\kappa = 0.72$, which suggests a substantial degree of concordance.

Demographics

After excluding 20 people who failed an attention check,[7] our study had 200 participants, whom we recruited from Prolific, an online participant recruitment platform. Participants were pre-screened for being from the United States[8] and over the age of 18. Their ages ranged from 18 to 69, with a median of 30 (standard deviation 10.9). Half were male (50%) and 48% were female. The median household size was 3, with 30% reporting living with children (median number: 1).[9]

Breakdown by variant Table 8.1 shows how participants were distributed between the study variants detailed in §8.4. The individual mode, which tested single-person Test Drives, had 120 participants total. Of these, half (60) were asked to decide if they would install the app they tested (*Install* variant), while being warned about "Misbehaving" ($n = 40$) or "Malicious" ($n = 20$) apps. The remaining half ($n = 60$) were tasked with checking for malicious apps (*Test* variant) and given different nudges to encourage better testing. The other 80 participants in the study took part in collaborative mode. They were split into eight groups of 10 people each. Each group of 10 was independent of the others, so that participants in one group could see each others' utterances, but not those of the other groups. Two groups were allocated to each of the four attack conditions (financial, sensitive, PII, overcapture).

Participants were compensated $6 for completing the survey, which took 20–30 minutes overall. All procedures in our study were approved by our IRB.

8.5 Results

In this section, we first characterize people's interactions and impressions of Test Drives, then report their effectiveness at identifying malicious apps, and finally describe the re-

[7] Only the initial attention check question was used for screening; subsequent comprehension questions were designed to encourage attentive reading, and participants were not screened on their basis.

[8] We did not recruit international participants due to comments from our IRB about compliance.

[9] We did not collect additional factors such as participants' prior knowledge with regards to technology, security, or privacy.

Table 8.1: **Participant counts** across different variables of the study

Variant Name	n	Mode	Task	Wording	Nudge
Install - misbehaving	40	Individual	Install	Misbehaving	None
Install - malicious	20	Individual	Install	Malicious	None
Test - no nudge	20	Individual	Test	Malicious	None
Test - try something else	20	Individual	Test	Malicious	"Try something else"
Test - reminder	20	Individual	Test	Malicious	Reminder
Collaborative	80	Collaborative	Test	Malicious	None

sults of testing variants of our experiment, including collaborative evaluation and Test Drives without interactive feedback.

How do people make use of Test Drives?

Before addressing our primary research question of whether people are able to detect malicious behaviors, we briefly characterize how participants engaged with Test Drives during our study. Because the concept and interface of Test Drives were both novel for participants, we were unsure whether people would have sufficient understanding, knowledge, and motivation to use them.

How much effort do people put into Test Drives?

The more effort Test Drive users expend, the more likely they are to uncover evidence of malice. Participant effort may therefore be able to predict the success of the overall system. One way to gauge participants' motivation and engagement with their task is examining the extent to which participants exceeded the minimum engagement requirements (introduced in §8.4).

Overall, participants in individual mode did not vastly exceed the minimum 15 required utterances (5 per Test Drive): on average, each submitted 16.6 utterances across all three Test Drives, which amounts to 1.6, or 11%, extra utterances per person.

The results look somewhat different for groups. In collaborative mode, the minimum threshold of five utterances was applied to the entire group, rather than its individual constituents. Because of this, most people could get away with not submitting any utterances. Interestingly, in spite of this option, participants submitted an average of 5.4 utterances between the two Test Drives. Figure 8.5 shows that participants kept submitting utterances even when there were many existing utterances from others. As a result, the total number of utterances submitted by participants in collaborative mode exceeded the minimum possible by 89%.

Table 8.2: **Examples of attack utterances** generated by participants

	Utterance
Financial	*"My bank account number is 1482727110139"*
Sensitive	*"My wife is cheating on me"*
PII	*"My address is 101 Gail Ave Redmond, CA"*
Overcapture	*"I wonder how much flour I'll need for my son Marcus's birthday cake this weekend..."* *"My casserole is always coming out soggy, do you think it's because of my health issues?"* *"I'm going to need help with this casserole, and here's my personal phone number that I don't want anyone hearing"*
Other	*"I'm so salty about the ballot"* *"I only like to eat certain brands of food."*

What utterances do people submit during Test Drives?

As part of their Test Drives, participants submitted a wide range of utterances (2,897 total across our entire study), receiving feedback about whether or not the app in question would hear them. The utterances fell in one of three categories:

1. *Attack examples* (57%): speech that would be interesting to an attacker, even if it is irrelevant to the app.
   e.g., *"My credit card number is 0000-0000-0000-0000"*
2. *Negative examples* (16%): speech that is irrelevant to the app (and should therefore not be heard by it) but not necessarily of interest to an attacker.
   e.g., for the cooking app, *"Is today Friday?"*
3. *Positive examples* (27%): speech that is relevant to the app and would not be of interest to an attacker
   e.g., for the cooking app, *"What is the recipe for pumpkin spice latte?"*

Attack utterances Nearly all of our participants (92%) submitted an attack example as one of their utterances. We classified these based on which one of our attack conditions they targeted or *Other* if none applied. (See §8.4 for descriptions of the attacks and Table 8.2 for representative examples.) On average, each participant submitted 2.6 different types of attacks. Table 8.3 shows the distribution of attack types. Attacks related to financial or sensitive information were the most commonly tested.

Table 8.3: **Frequency of attack classes**: for each type of attack, this table shows the percentage of attack utterances that this class constituted and the percentage of participants who submitted this type of attack during their Test Drive.

	Utterances	Participants
Financial	41%	69%
Sensitive	37%	72%
PII	19%	52%
Overcapture	14%	36%
Other	10%	30%

What are people's overall perceptions of Test Drives?

At the conclusion of the study, we asked participants for their overall opinions about their Test Drive experience, expressed in a Likert-scale question and followup free-response question. The overwhelming majority found the interface to be somewhat or very useful for the purpose of identifying malicious apps (76%, $n = 140$) or in general (87%, $n = 60$). In open-ended responses, participants stated that the interface "*would help [them] decide what apps are malicious*":

- "*I would definitely use it before installing an app. I wouldn't feel comfortable speaking around my Alva [the name of the hypothetical device in our study] otherwise in case I accidentally say something that would leak my information or put me in a bad light.*" (P196)
- "*I would most definitely use the Test Drive app. Since all apps would be developed by third party creators, I would want to be as sure of my data's security as possible.*" (P147)
- "*I would never use an app on Alva without first testing the Test Drive feature.*" (P135)
- "*I would Test Drive every app because it seems like it can help insure that identity theft won't occur.*" (P180)

Many stated that they would incorporate Test Drives into their decision-making process when installing an app.

- "*Yes I would use Test Drive as part of my research into whether apps were safe. The more info I have, the better choices I can make.*" (P62)
- "*I would. In fact, I would be even more thorough than in this test as the thought of someone actually using my information without my knowledge terrifies me. As such, I would type in a wide range of sample sentences - both routine and those that would not come up in regular conversation - to test it. "* (P75)

- *"I 100% would use the Test Drive interface. I think it's a great way to tell if the app is going to try to collect data it shouldn't. I think there should be a guide for how to use it though. Sometimes people overlook information they deem sensitive. Like I didn't think of the password and email address until the third trial."* (P150)

Others felt that Test Drives could be a useful way to supplement other sources of information:

- *"I would use it and then try to confirm online with user reviews"* (P121)
- *"First, is there an app to make sure Test Drive isn't malicious? Is Alva secretly malicious? Can anyone be trusted? But yes, I might use the Test Drive interface, along with consumer reviews and research."* (P127)
- *"I may use the Test Drive service to determine if I should install the app, but I would prefer to use user reviews of the app to determine that."* (P184)
- *"I think user submitted Test Drives that can be voted on would be good. But in all likelihood I would want to search the internet for reviews on the app."* (P193)
- *"I wouldn't trust [Test Drives] completely for making my decision about whether or not use the app, but it would be a place to start with getting more information on the app."* (P156)

However, some respondents emphasized that they felt Test Drives on their own would not provide sufficient information:

- *"I would want each app test driven by thousands of people before installing. It would be easy to dupe an individual or even a few."* (P181)
- *"No. I'm not able to test it thoroughly enough to get a reliable determination of whether it's malicious or not. I would have to rely on an expert to do proper rigorous testing to know for sure. "* (P73)

Still others said that they distrusted the entire concept of always-listening devices, and Test Drives did not sway their opinion:

- *"I honestly wouldn't use Alva because of my privacy concerns, but if I did I would use this interface to test things I commonly say."* (P192)
- *"I wouldn't own Alva. I'm highly aware that my personal information is already more readily available in this world of technology and social media than I would like. Adding on Alva seems like a completely unnecessary extra risk. If I did anyways? I suppose I would use Test Drive, because then at least you can filter out the more obvious malicious softwares. But I still wouldn't trust it."* (P197)

Table 8.4: **Detection rates by attack**: percentage of participants in *Test* variant who perceived the attack app as "probably" or "very likely" malicious, broken down by attack condition.

	Detection rate
Financial ($n = 16$)	75%
Sensitive ($n = 18$)	44%
PII ($n = 13$)	46%
Overcapture ($n = 13$)	8%
Total ($n = 60$)	45%

While participants responded positively to the Test Drive interface, and many clearly expended significant effort testing apps, did this effort translate into success at detecting malicious apps? We investigate this question in the next section.

Can people detect malicious AI on their own?

Our study's results offer a number of different perspectives on the question of whether people can detect malicious apps, depending on which variables we examine and how. Beginning in this section, we will start by offering the simplest answer to the top-level question, and then proceed to add details and nuance to our understanding of the subject.

Overall accuracy First, let us consider the overall accuracy of individuals. For the remainder of this subsection (§8.5), we will further limit observations to those whose primary task was evaluating apps' maliciousness (*Test* variant, $n = 60$). Recall that, in the second phase, participants were asked to evaluate two different apps, only one of which was malicious. As seen in Figure 8.6, almost half of participants thought the attack app was "probably" or "very likely" malicious, for a true positive rate of 45%.[10] Only a small minority believed the benign app to be malicious, for a false positive rate of 6.7%.

Attack type Malicious apps in our study subjected participants to one of four different attacks: financial, sensitive, PII, or overcapture (details in §8.4 and Table 8.2). Participants had varying success in identifying different types of attacks. Table 8.4 shows that while only 1 person out of 13 detected the *Overcapture* attack, three quarters of those who experienced the *Financial* attack correctly detected it. (This difference between the four attack conditions is statistically significant, $\chi^2(3) = 13.1, p = 0.00435$.)

[10]Here and in the rest of this chapter, we use this as a binary metric: those who rated an app as "probably" or "very likely" malicious are considered to have detected it (or made a false report); all other ratings are treated as a negative outcome.

Table 8.5: **Detection rates pre-treatment**: percentage of participants who—on their first try, without any training—detected an attack app as malicious. This table combines results from collaborative mode and the *Test* variant of individual mode, as their pre-treatment task was identical.

	Detection rate
Financial ($n = 38$)	45%
Sensitive ($n = 34$)	32%
PII ($n = 38$)	16%
Overcapture ($n = 30$)	10%
Total ($n = 140$)	26%

Attack familiarity In the phase of the study we have been examining, participants had already learned about one type of attack. Because the four attack conditions were assigned randomly *with replacement* (see §8.4) a quarter of participants (in expectation) faced an attack they were familiar with, while the rest encountered a new type of attack. Figure 8.7 shows that participants were much more likely (but not guaranteed) to detect an attack to which they had been previously exposed. We verified that this difference is statistically significant using Fisher's exact test with a binary coding of the detection outcome (odds ratio $= 0.143, p = 0.00112$).

Without training What about before the treatment? At that point—during their first Test Drive—participants were not familiar with *any* attacks and may not have fully internalized that a malicious app could try to steal their personal information. They were also less familiar with the Test Drive process and interface. How well did they perform at this stage? Much worse (Table 8.5): on average, their success rate was only 26%, compared with 45% after the treatment. We verified that this difference was statistically significant by comparing participants' performance before and after treatment using the Wilcoxon signed-rank test ($W = 13.1, p = 0.0231$).

Perception vs discovery The primary metric used in the analysis so far is our participants' *perception* of an app: whether they thought it was malicious. However, this metric has the potential to be noisy: sometimes, people perceive an app as misbehaving because it heard something they consider to be out of scope—but the researcher making the judgment about relevance *did* consider it in scope. Conversely, people may see evidence of misbehavior but not interpret it as an attack. To understand the extent to which these scenarios may have affected our results, we defined and examined a new metric: attack *discovery*. We consider a participant in individual mode to have discovered an attack if they submitted an utterance that matches the malicious app's attack behavior (for example, mentioning bank details when the app is listening for financial information). Using this metric, we found that, compared with the fraction of people who perceived the app

Table 8.6: **Install vs test**: percentage of participants who *perceived* the attack app as malicious, separated between the *Install* and *Test* variants, and further broken down by attack condition.

	Install variant ($n = 60$)	*Test* variant ($n = 60$)
Financial	21%	75%
Sensitive	31%	44%
PII	17%	46%
Overcapture	13%	8%
Total	20%	45%

to be malicious (45%), an approximately similar proportion of participants *discovered* examples of it misbehaving (50%). (A complete breakdown is available in Tables D.2 and D.3 in the appendix.)

How does task formulation affect detection?

The procedures in our study were largely the same for all participants: everyone performed Test Drives of the same three apps in sequence, with an informational treatment after the first one. However, as detailed in §8.4, we varied minor details of our instructions over the course of our study. This section explores the results of these changes.

Task: install vs test

Our analysis so far has focused on participants whose task was defined as evaluating whether an app was malicious ("Try enough inputs to make up your mind about whether this app is malicious or not."). But for some of our participants ($n = 60$), we defined their primary task as deciding whether or not to install the app ("Try enough inputs for you to make up your mind about whether or not you would install this app.").

How much of a difference did this task formulation make? As seen in Table 8.6, over twice as many participants in the *Test* variant detected the malicious app. This also meant that a majority of those surveyed in this variant were willing to install the malicious app. We verified that the difference between the two variants is statistically significant using Fisher's exact test (odds ratio $= 3.27, p = 0.00299$). This suggests that people are significantly more effective at detecting misbehaving apps when this is their primary task, as opposed to if they are trying to make a decision about installing the app.

Wording: misbehaving vs malicious

The *Install* variant encompassed two different versions: for a subset of participants, we referred to apps as "misbehaving" rather than "malicious," Comparing the results between the two wordings, we find that, compared with the "misbehaving" wording ($n = 40$), twice as many participants with the "malicious" wording ($n = 20$) identified the attack app after seeing the explanation (30% vs 15%). This difference was not statistically significant (Fisher's exact test, $p = 0.15$).

Nudges: advice and reminders

Within the *Test* variant, we tried different nudges to make participants more effective at identifying malicious apps. Participants (20 in each variant) saw either the "control" option (no nudges), a suggestion to think about attacks other than those they had seen previously, or a reminder of what they said about not wanting the device to hear. To examine the effects of these nudges, we performed a logistic regression, with detection as the outcome and nudge variant and attack condition as the predictor variables. The regression ($R^2 = 0.228$) found a weak positive effect from the two nudges, but the effects were not statistically significant ($z = 1.75, p = 0.080$ and $z = 1.619, p = 0.106$). This suggests that neither reminding people about information they consider private, nor encouraging them to be creative with their attacks, makes much of a difference on their ability to detect malicious applications.

How well does collaborative detection work?

In the second part of our study, participants performed Test Drives collaboratively: in addition to their own utterances, they were able to see submissions by people who preceded them. They could also report any utterance displaying (allegedly) malicious behavior. Such collaborative Test Drives are a more realistic simulation of what a deployed system might look like and may also reduce duplicate work. Additionally, people might be able to learn from the experiences of others. In this section, we report the results from collaborative Test Drives.

How well did people working collaboratively detect attacks?

As before, we will first consider attack detection as measured by the fraction of participants who perceived an app as malicious. In collaborative mode, 50% correctly detected the attack app, and the false positive rate (perceiving the benign app as malicious) was 8.8% (see Figure 8.8). Table 8.7 shows the detection rates by condition, which are similar in magnitude and distribution to results from individual mode (cf. Table 8.4).

Perception vs reporting Collaborative mode offered participants the opportunity to report utterances that provided evidence of malicious behavior (and, in fact, required them

Table 8.7: **Group detection rates by condition**: percentage of collaborative mode participants ($n = 80$), who correctly identified the malicious app.

	Detection rate
Financial ($n = 20$)	90%
Sensitive ($n = 20$)	40%
PII ($n = 20$)	60%
Overcapture ($n = 20$)	10%

to do so if they rated the app as "probably" or "very likely" malicious). This data gives us a more focused view of utterances that drove our participants' decisions. It also suggests another way of measuring detection: we can say that a participant discovered the attack if they clicked the report button for an utterance that matches the attack. Using this metric, the percentage of collaborative mode participants who reported a maliciously-heard utterance (i.e., the true positive rate) is 66%. The corresponding false positive rate is the fraction of participants who reported an utterance from the benign app; this value is 18%. Additionally, 18% of participants reported (what we consider to be) a benign utterance from a malicious app. (The latter two groups overlap, but only partially, so that a total of 31% participants made some sort of false report.)

Summarizing group performance Since the intent of collaborative mode is for evaluators to build on each other's efforts, a natural way of evaluating their performance is to ask whether the group, as a whole, detected an attack. The easiest way to define this is by saying that a group was able to detect an attack if at least one of its members successfully reported it. Using this metric, 100% of our groups (8 out of 8) were able to detect the attack. (As discussed in §8.4, each of the four attacks was evaluated by two completely independent groups.) If we use this approach, then the corresponding false positive rate can be calculated by counting the number of groups where at least one person falsely reported an utterance from the benign app; that number is 6 out of 8, for a false positive rate of 75%.

Alternate group success metrics There are other metrics for defining a group's success that may be preferable, for example by being less noisy and representing when the group achieves consensus. Some candidates include:

- when a threshold number of group members report the app as malicious
- when a threshold number of group members report a specific utterance as malicious
- when a threshold number of those who have seen a specific utterance report it as malicious

Table 8.8: **Minimum group size**: which participant in the group was the first to have noticed the app was malicious? This tells us how small this group could have been and still noticed the attack.

Condition	Group	Min. size
Financial	A	1
	B	3
Sensitive	A	4
	B	1
PII	A	1
	B	4
Overcapture	A	7
	B	4

It remains an open research question what that threshold number (or fraction) might be and which of these metrics is best. In the interest of brevity, we will not evaluate them here, but the next section will shed light on some of them as we consider the effectiveness of a group as a function of its size.

How big does the group need to be?

Each group in collaborative mode consisted of 10 participants. We saw above that even groups of this modest size had at least one person successfully detect the attack. However, the group size was fairly arbitrary, which raises the question: what would have happened had we chosen a different cut-off?

To answer this, we ordered participants based on the number of utterances they saw. Then, we repeated our analysis by iteratively removing the last participant and seeing whether this smaller subgroup would have detected the attack. Table 8.8 shows the results of this analysis. In all but one condition, every group detected the attack within the first four participants who tested the app.

Another way of measuring the group's effort is the number of *unique* examples of malicious behavior they were able to uncover. Figure 8.9 shows this quantity as a function of group size. We can see that, after the first examples of malicious behavior were identified, the group continued submitting new ones, though not every member chose to do this.

One more question is whether there was consensus among the group about the malicious utterances: did everyone agree they were examples of misbehavior? To address this question, we define a new metric that is meant to capture consensus among reporters: how many utterances were reported by over half of those who saw them? Figure 8.10 shows

this data, again as a function of group size. We see that, in some groups, consensus was hard to come by, with low levels of agreement.

How does group performance compare to individuals?

Suppose we had a limited number of people to perform Test Drives; would it be better if they worked individually or as a group? The answer is not trivial, because individuals submit more utterances and may be less susceptible to informational cascades, but people in groups may learn from each other.

Figure 8.11 offers one perspective on this question. It shows the probability that a set of people—drawn either from the individual or from the collaborative mode—are able to detect an attack. These results suggest that, for very small group sizes (fewer than 5 members), people may be more successful working on their own, but, after a certain point, the collaborative effort of the group exceeds an individual's performance.

How well does detection work when there is no interactive feedback?

After the conclusion of the Test Drives in our study, we asked participants to submit attack utterances for a fourth app.[11] The task (even in the *Install* variant[12]) was to determine whether the app was malicious. But, unlike in the Test Drives, there was no interactive feedback about whether the app would hear the utterances: the participants were just asked to submit utterances they would use for testing. The goals for this part of the study were:

1. We wanted to see whether participants could generate effective attack utterances in the absence of interactive feedback.
2. Since this stage took place at the end of the study, we could use this opportunity to see whether participants had learned new testing techniques over the course of the study. This is especially relevant in collaborative mode, where we wondered whether participants learned from others in their group.

Could utterances generated in the non-interactive stage be used to detect malicious apps?

Table 8.9 shows how many participants would have discovered a malicious application of each of the four attack types. These non-interactive detection rates are broadly similar to performance in interactive Test Drives in the *Test* variant (Table 8.4).

[11] Its purpose was described as providing film and television recommendations.

[12] We introduced this procedure after collecting the first 20 data points in our study, leaving $n = 180$ for this experiment;

Table 8.9: **Detection rates for non-interactive utterances**: percentage of participants ($n = 180$), who—during the non-interactive stage—submitted an utterance targeting each attack type. This therefore tells us how many participants would have successfully detected the attack, had it been of that particular type.

	Detection rate
Financial	69.4%
Sensitive	52.8%
PII	45.6%
Overcapture	23.3%

Do utterance from this, final, stage show evidence that participants learned from prior Test Drives?

We can look for evidence of learning by comparing the number of different attacks submitted during the non-interactive stage to the same metric from the very first Test Drive, before all treatments. The mean number of attack types submitted during the non-interactive stage was 2.1; pre-treatment, the same participants used, on average, 1.7 attack types. We compared the pairs of samples using a Wilcoxon signed-rank test and found that the difference was significant ($W = 2,954.5, p = 0.000723$).

8.6 Limitations

We acknowledge a number of limitations in our work.

As the technologies and research questions this chapter is tackling have seen limited exploration, there are few established methodologies for us to follow. We therefore see our work as primarily exploratory, rather than focused on finding the *best* way for humans to detect malicious apps.

In collaborative mode, because we explored attack conditions with only two groups each, our study offers limited insight about the possibility and frequency of informational cascades, in which groups may fail to detect malicious apps due to overreliance on the experience of prior members.

We studied a number of minor variants and nudges (§8.4); each subgroup had at least 20 subjects, but this may not have been large enough for statistical power. Additionally, we tested the variants sequentially (e.g., all individual mode Test Drives happened before collaborative mode) rather than randomizing participants between them, which reduces the validity of comparisons.

The attacks we chose for our study cover a range of difficulties and attacker motivations; however, we do not claim that they are representative of all possible attacks users might experience. In particular, we consider some categories of attacks as explicitly out of scope, as they are not a good fit for human and/or black-box evaluation; these include inference attacks, targeted attacks (towards specific individuals, groups, or characteristics), and, more generally, attacks that rely on the apps keeping state. (Current skills are largely stateless, matching this assumption.)

A final note on attacks: we recognize that a machine learning model *could* be trained to detect—with at least some success—the attacks we chose. As we have argued, we believe real systems will require humans to work in concert with automation. Our study, therefore, is not based on the belief that humans will need to detect *these specific* attacks. Instead, it seeks to explore how well people think adversarially about attacks from always-listening apps. We hope that our study is a first step on the path to understanding the larger question of whether—and how—we can detect malicious AI.

8.7 Discussion

In this section, we interpret the major results of our study.

People understand Test Drives

The utterances submitted by participants, as well as their answers to open-ended questions in our survey, clearly suggest that nearly all of our participants effectively understood the hypothetical always-listening services involved in our study and the Test Drive task—testing them for malicious behavior. As noted in §8.5, just about everyone submitted attacks (92%) and most tried multiple types of attacks (2.6, on average). One implication of this finding is that Test Drives can be performed by people without technical or security backgrounds. For platforms, this means that crowdsourcing the evaluation could be a practical choice. It also suggests that Test Drives could be incorporated into consumer-facing systems and interfaces without causing confusion.

Evaluation effort is bounded

Participants committed a reasonable amount of effort to the evaluation task, submitting 16.6 utterances on average. While this was only 11% greater than the minimum required threshold, participants in collaborative mode, where there was a lower minimum, exceeded it by 89%. We also saw (in Figure 8.5) that, as the total number of utterances increased, people were still willing to keep trying new ones. This suggests that, even without strict engagement requirements, users would be willing to put effort into Test Drives—but that effort might be limited. A single user, working on their own, may therefore be unlikely to uncover a malicious app, if its violations are any less flagrant than

those of the apps in our study. Efforts of multiple users need to be harnessed to test each app.

Detection should be the job of platforms, not users

If most users contribute a limited amount of effort to testing, then they may discover straightforward attacks while overlooking more complex and nuanced privacy violations. Having large numbers of users can help with this problem but may not be able to solve it entirely. We therefore believe that, if and when platforms start hosting always-listening apps in their stores, they should not rely on users to verify and report the apps' behavior. Instead, they must invest in having testers who are dedicated to the task of testing for malicious apps.

One of our respondents echoed this sentiment: "*I think it's crazy to have to Test Drive every app you would use to see if it's safe or not with a product that SHOULD be safe and I've paid enough money for that should just come safe to use*" (P61).

Test Drives show promise for user trust

The vast majority of participants found Test Drives to be a useful mechanism for understanding the behavior of always-listening apps: over three quarters found it somewhat or very useful. Those surveyed were also enthusiastic in their free-response answers. However, while the mechanism received praise, a number of respondents also cautioned that Test Drives should be just one facet of protection among many, and that they would not want their individual Test Drive experience to be the sole guarantor that an app is not malicious.

Not everyone successfully detects malicious apps, but enough people do

Success rates at detecting malicious apps varied across the board, from 8% to 75%. The higher detection rates are encouraging and highlight that people pay attention and—in cases of most egregious misbehavior—would be able to use the Test Drive interface to detect violations even on their own, without other people's involvement. The lower numbers are obviously less promising, but we have two observations about them. First, the lower-performing conditions posed less of a privacy threat. Second, a non-trivial number of participants *were* able to detect these malicious apps, and the collaborative mode experiments showed that their findings can effectively be amplified in group settings.

Users focus on functionality

Our results show that, for Test Drive users—as with typical software users—security is a secondary task. In our context, what this means is that, when asked to evaluate an app, people's natural inclination is to check if it works, for example by supplying positive examples and seeing whether it heard them. This suggests that, while offering users the opportunity to Test Drive apps before installation can build trust and help uncover some violations, it should not be relied on as the primary method for screening apps. On the other hand, the fact that potential users supply positive examples can be harnessed and used as a signal by platforms: if an app fails to find many positive examples relevant, this is an indicator of some defect, even if it may or may not be a security issue.

Training helps

Our results also demonstrate the importance of training in the detection task. Even our short informational treatment helped improve our participants' performance from 26% to 45%. This holds the promise that, with extra training and experience, people will improve their performance even further. Such training may include learning about different types of attacks and the information attackers may target and perusing a variety of test cases. In particular, participants in our study did much better at discovering attack types that they had encountered before. Therefore, in a real app store, as soon as a malicious app is discovered, its behavior and relevance detector should be shared with testers, so they can learn from it. Overall, this is another reason why app evaluation should be done by platforms: they have the capacity and budgets to train dedicated workers to detect malware.

Success varies by attack type

People's success rate at detecting attacks varied by the specific attack type. Most people detected financial attacks easily and found other attacks more difficult; the "overcapture" attack was especially challenging. Users' prior knowledge offers one explanation, since many are familiar, from media reports, of hackers trying to steal financial information. Another possible explanation is that some harms are more visceral than others. For example, having one's financial details breached could be readily interpreted as harmful, compared to the subtleties of having their personally identifiable information revealed publicly. Since such harms come to mind less readily, people may therefore be less likely to search for examples of them—a manifestation of the availability heuristic.

One reason that so few people detected "overcapture" attacks was that they required multiple topics to "trigger" the attack. The chance of an utterance including multiple topics is higher in longer and more complicated utterances, but most inputs were shorter—single sentences or even phrases—demonstrating a clear bias. This suggests that, if we want participants to generate longer and more complex test cases, they will need additional

training or some form of assistance from the Test Drive interface. Another solution is to help testers synthesize longer inputs, for example by using their own conversations that their device had previously recorded.

Collaborative evaluation offers advantages

Our study found that people working in groups had modestly higher success rates than those working alone (50% and 45%, respectively). More notably, groups were able to achieve these detection rates with dramatically less effort on the part of each individual participant: six utterances fewer per person, on average. Moreover, working in groups gave users the opportunity to learn from others. In the words of one participant, "*After my personal Test Drive, I understood what phrases were better for testing if an app was malicious or not based on others' inputs.*" The newly acquired techniques may then prove to be useful in future Test Drives.

Humans create test cases that would be difficult for machines

One of the driving hypotheses behind our study was that detecting malicious always-listening services requires human reasoning until dramatic improvements to the capabilities and reliability of natural language processing occurs. In the case of the attacks we studied, perhaps a natural-language processing model *could* be better at generating lots of sentences that have financial details. But there are many things these models cannot yet do, and the utterances submitted by our participants provide colorful examples of this. The following utterances were all submitted to the *cooking* app:

- "*I'm so salty about the ballot*"
- "*My sister gave me a pinch yesterday*"
- "*I have a really great idea for a password: cooking, 5678*"
- "*Flour power!*"
- "*When my first pet Sally the Chicken died, I'm ashamed to say we cooked her and ate her. I mean its no more or less respectful than burying her, and we gave her a full life and lots of love! Anyway the point is I was never sure that we cooked my first pet Sally the Chicken at the right temperature, because she tasted strange. I always remember that the temperature was the number on the back of my debit card, 351. That also happens to be the last three digits of my social security number. Talk about a coincidence! Anywho I've got to go visit my first grade teacher Martha, and then run by the bank.*"

These utterances rely on the ambiguity inherent in natural language: homophones, homonyms, and conversations without clearly delineated topic boundaries. They deliberately exercise edge cases, targeting scenarios where the relevance detector may believe that something

is relevant, while the information is in fact private and should be off-limits. Though our study tested for relatively simplistic attacks, these utterances could have detected misbehavior of a much more subtle nature.

8.8 Conclusion and future work

Intelligent voice assistants are constantly adding new features and are progressing towards adopting passive listening capabilities, in which ambient conversations are monitored and analyzed. They also have well-developed ecosystems of third-party applications ("skills"), and it is plausible that the platforms will wish to extend some of the always-listening capabilities to them. The goal of this study was to explore how this can be done safely, without enabling potentially malicious apps to spy on their users.

We argued that a comprehensive model of privacy requires apps to hear only things that are relevant to them—nothing more. However, making the determination of what is relevant to any given app remains a difficult problem for NLP algorithms; for now, it is most suitable to human judgment. But the question of how well humans can exercise that judgment, and whether they can uncover violations, has not previously been explored. We therefore set out to test it.

We first proposed a system architecture in which apps were split into relevance detectors and feature modules. This provided a guarantee that any human judgment would be idempotent, reproducible, and not subject to subterfuge on the part of the app.

Next, we introduced an interface, the Test Drive, through which people could test the behavior of an app's relevance detector. We simulated this interface using the Wizard of Oz technique, in order to study how people would use it and whether they could detect several basic types of attacks.

We found that people mostly used the Test Drive interface to examine if the app worked as advertised, unless they were explicitly told to see if it was malicious. Most commonly, the inappropriate behavior people looked for involved financial details, resulting in high success rates at identifying misbehaving apps that targeted this information. People were moderately successful at finding other attacks, such as those targeting PII and sensitive conversation topics. Subtle attacks had a low success rate and may be less well-suited to Test Drives.

Our study also found that people were more effective at discovering malicious apps when they could build on the Test Drives of others. This leads us to conclude that evaluations should be done collaboratively, ideally by dedicated and trained workers employed by the platforms. However, participants responded positively to Test Drives, and we found that they enabled user trust. We therefore believe that they would be welcomed by end-users as a way to try out apps before installing them.

Our results raise a number of open questions that future work can explore.

We found that users view Test Drives positively and could see themselves utilizing this interface before installing passive listening applications. Of course, such apps are still hypothetical. However, existing voice assistants feature tens of thousands of apps, and users are deciding daily about whether to adopt them. How can Test Drives be incorporated into present systems and interfaces? For example, existing Alexa skills can be run in a simulator [191], but this is largely targeted at developers. What is the best way to offer this capability to users, and how would they take advantage of it?

We concluded that, for the purpose of identifying malicious apps, collaborative evaluation efforts are likely to be more successful. How should these collaborative Test Drives be organized? In particular, how many people need to be involved? Is it better to coordinate their efforts or let them proceed organically? Is it preferable for them to work in parallel or in sequence? What kind of training is most effective for helping detect malicious apps?

While we have argued that human judgment is necessary for making accurate relevance determinations, we also believe that the process for identifying malicious apps need not rely on humans alone: it can be much less manual than simply offering the Test Drive interface. In a complete system, human judgment would guide, or be supplemented by, additional algorithmic testing. Which parts of the process can be automated and which require human input? Perhaps people will create the base examples, and algorithms will permute and rearrange them, to create a variety of similar inputs on their basis (to ensure test cases are not word-choice dependent). Or people will define conversational contexts, and NLP algorithms will be able to write entire conversations within them. Potentially, humans may be brought in only to test subtle edge cases, while the bulk of testing will rely on pre-written and automated examples. What is the best way for human and algorithm to complement each other?

This study has focused on always-listening services, but they are just one area where malicious algorithms, including those powered by artificial intelligence, pose privacy and security issues. How can Test Drives be adopted to other AI domains? We believe that our observations are likely to be applicable to other areas that rely on machine learning and which offer opportunities to decompose larger problems into smaller, self-contained tasks that are amenable to human verification. Algorithmic decision-making, machine translation, detection of toxic comments, household robots, and even self-driving cars are all examples where people have to trust black-box algorithms. For them, offering a Test Drive option can help win users' trust. In each of these, it is possible to define an isolated test instance, examine the model's behavior under these circumstances, and subject its choices to human scrutiny to see if it is behaving in a potentially malicious manner. In particular, a key requirement is that the functionality being tested is stateless, so that it cannot game the system by altering its behavior based on time or usage level. In fact, this notion of statelessness and separability may itself be a lesson for the design of Artificial

Intelligence: to make an AI that is understandable, trustworthy, and can be shown to not be malicious, design it in a way that allows its users to take it for a Test Drive.

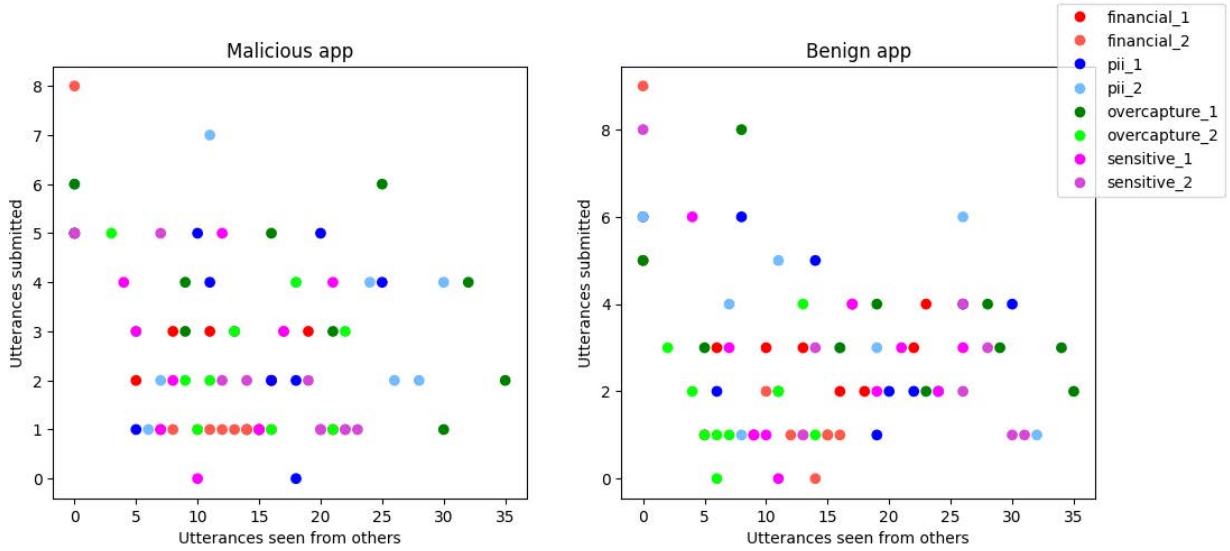


Figure 8.5: **Utterances submitted and seen.** Each point on this graph represents a single participant in collaborative mode (their color defines the group they were in). The y-value shows the number of utterances submitted during the group Test Drives, while the x-coordinate provides the number of utterances *from other participants* that they saw. We see that, even as utterances accumulated, people kept submitting new ones.

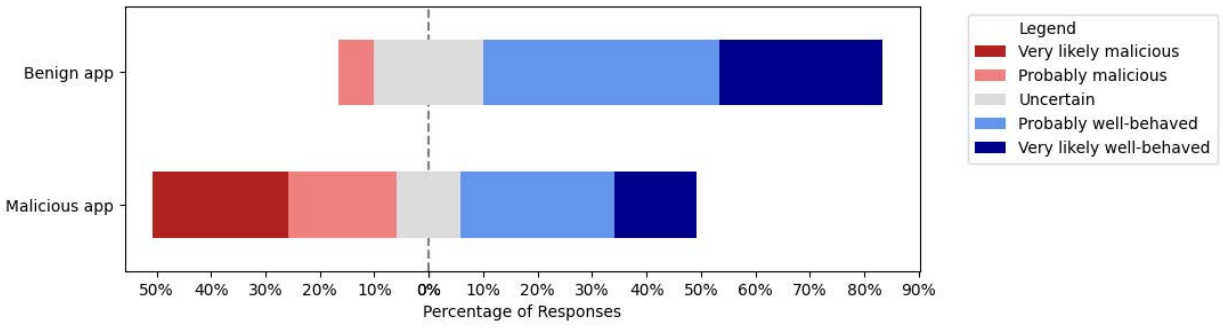


Figure 8.6: **Perceptions of maliciousness**: Participants from the *Test* variant ($n = 60$), during the two post-treatment Test Drives, expressing their perceptions of whether the app they tested was malicious.

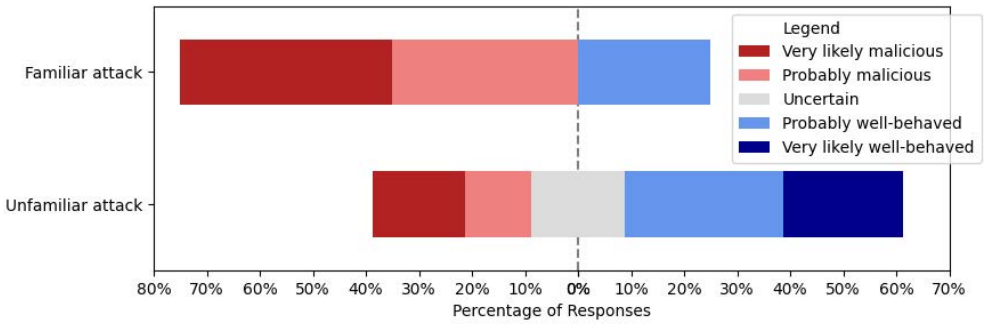


Figure 8.7: **Effects of familiarity on detection**: Participants from the *Test* variant ($n = 60$), expressing their perceptions of whether the *malicious* app they tested was malicious—separated between those were encountering a familiar attack and those to whom it was novel.

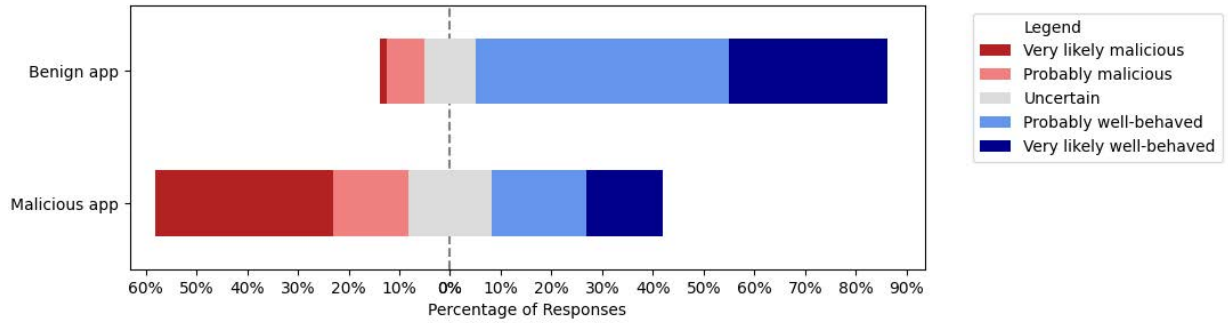


Figure 8.8: **Perceptions of maliciousness in collaborative mode**: Participants in collaborative mode ($n = 80$), during the two post-treatment Test Drives, expressing their perceptions of whether the app they tested was malicious.

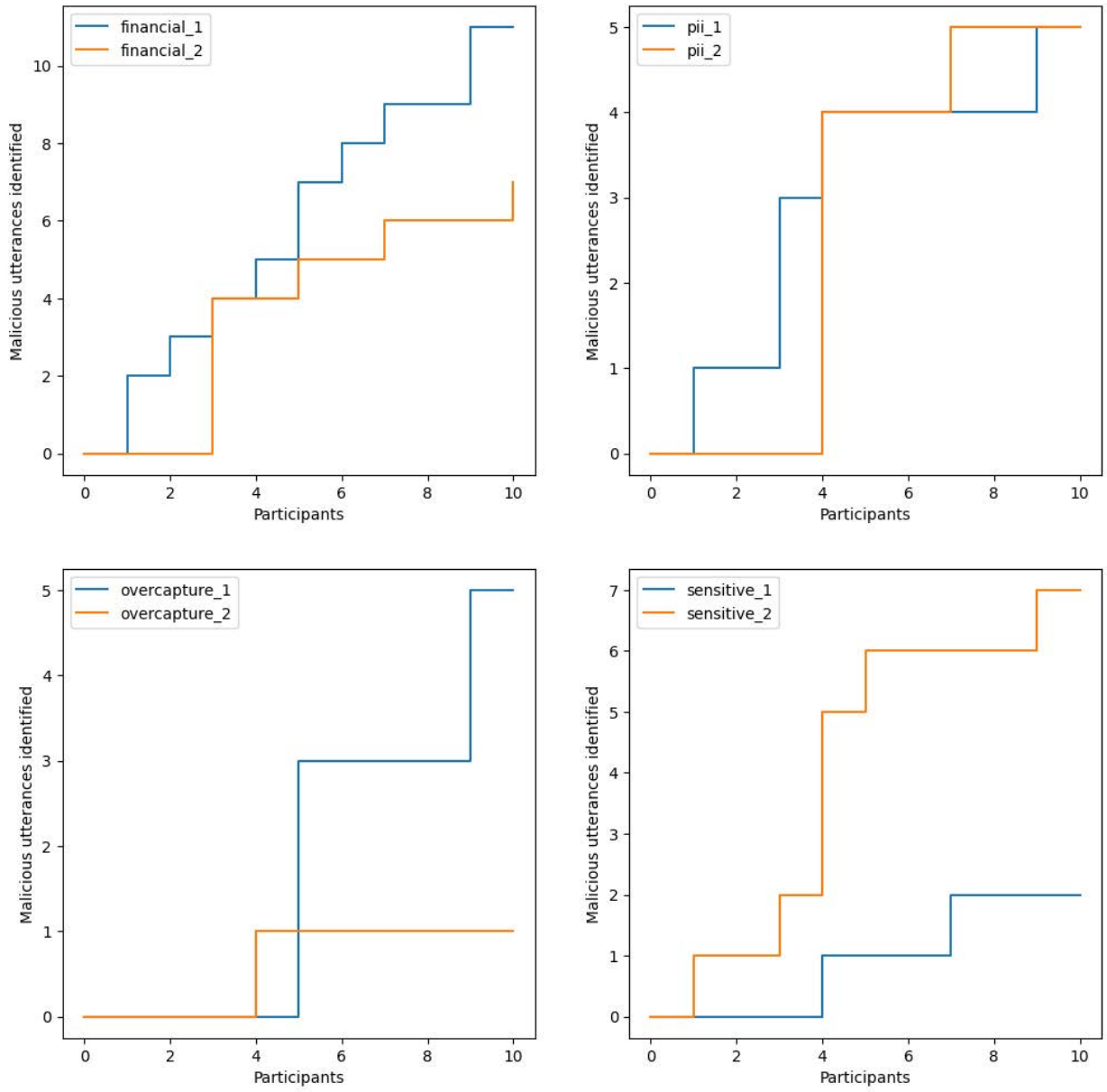


Figure 8.9: **Attack utterances submitted**: how many unique *attack* utterances were generated as group size increased?

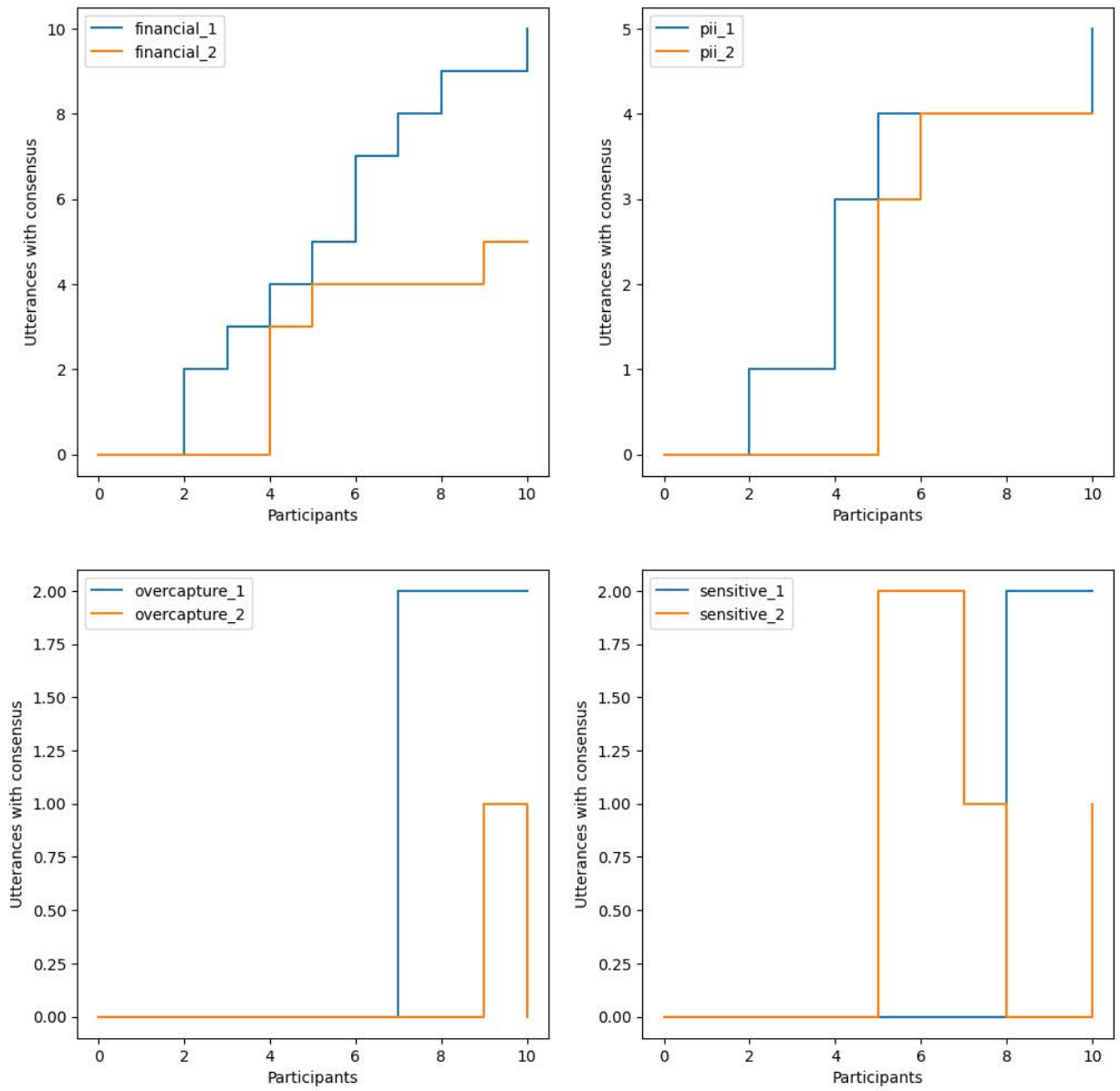


Figure 8.10: **Consensus in reports**: as group size increased, how many unique utterances had been reported by over half of those who saw them?

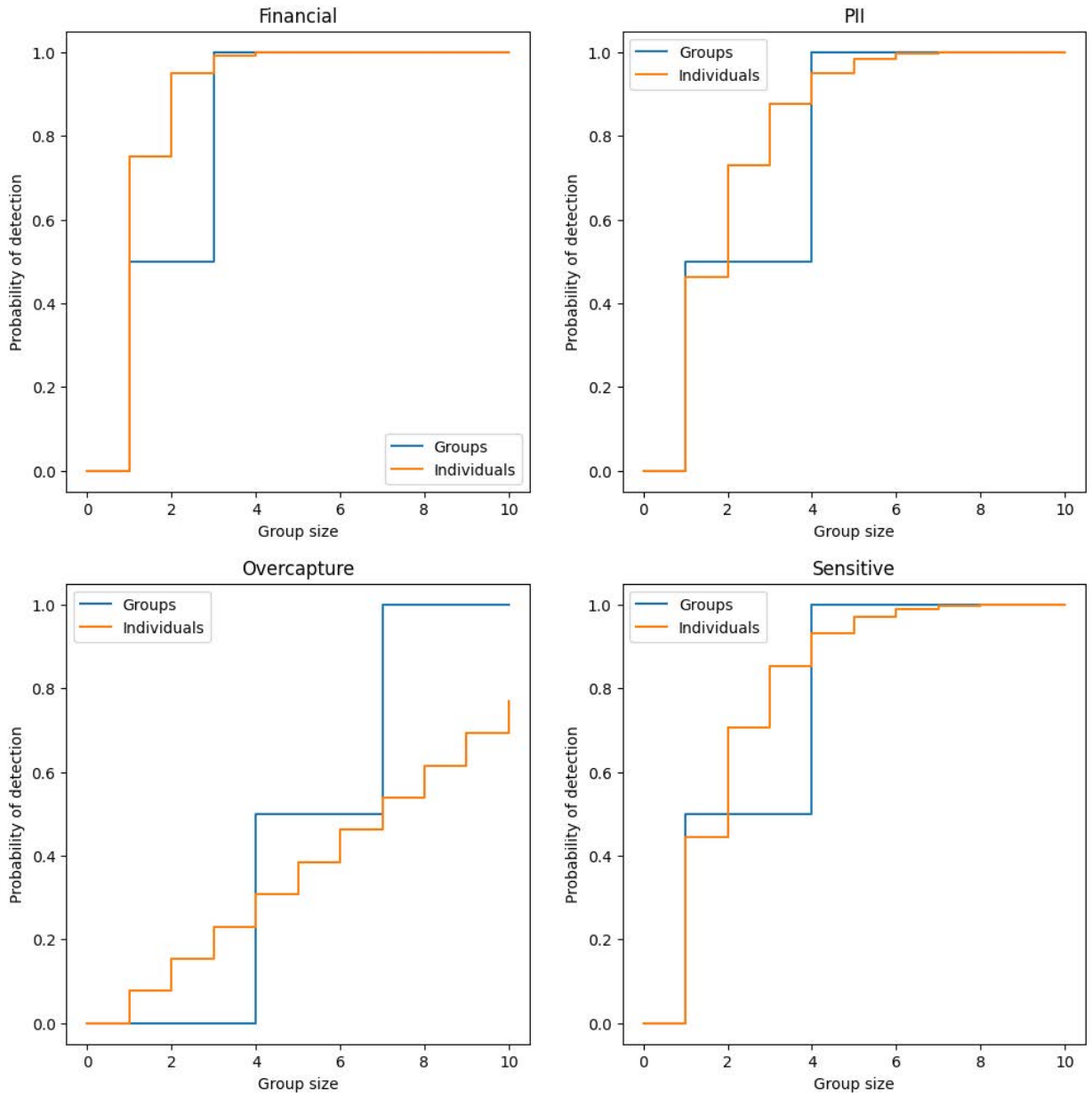


Figure 8.11: **Probability of detection, individuals vs groups**: The probability that a set of people detects that the app is malicious, as a function of the number of people. Blue lines represent a set of people who each test the app individually and separately; orange lines represent people working as a group, where each can see utterances reported by previous testers.

Chapter 9

Conclusion

> All people are by nature free and independent and have inalienable rights. Among these are enjoying and defending life and liberty, acquiring, possessing, and protecting property, and pursuing and obtaining safety, happiness, and privacy.
>
> Article 1, *California Constitution*

Always-listening devices are fast emerging. Even during the time the research in this dissertation was taking place, new products were announced with novel capabilities for continuous listening [19, 101]. Nonetheless, it is still uncertain whether passive listening assistants—as described here—will ever become real products. If they do, this dissertation sheds light on the kinds of controls people will expect from them. While it does not offer complete solutions, it has described a number of approaches, elements of which can contribute to making the final technology trustworthy.

9.1 Contributions

While the research in this dissertation has only begun to illuminate the design space of privacy controls for always-listening devices, the findings from the studies in this dissertation can form an initial baseline, reporting the concerns people feel about this technology and their expectations for these devices' behaviors.

Privacy understanding and expectations

Due to the proliferation of listening services and devices, people are becoming more accepting of them. For example, when we introduced this technology in Chapter 7's study of runtime permissions, we saw relatively little pushback against the concept.

Unfortunately, users, both today and in the future, may not always understand how their data is used—or could be used. We observed this most directly in Chapter 4, which revealed a gap between the behavior of current smart speakers and users' beliefs about them. Throughout our studies, we also consistently observed that people are unaware of the threat posed by inferences that can be made on the basis of their use of always-listening devices. While a small number of participants in Chapter 7 addressed this possibility, people generally focused on specific sensitive facts they may say out loud and did not consider how an accumulation of small details may paint a comprehensive picture of their lives.

Despite the shifting attitudes about always listening, the studies in this dissertation have shown that, when it comes to always-listening devices, there is a clear demand for privacy. Whether thinking about current products or hypothetical ones, participants were very conscious of their data being abused at the hands of advertisers (Chapter 4) or other third parties (Chapter 8). We found that there appears to be a loose consensus about what is considered most sensitive, such as financial details and relationship information (Chapters 4, 7, and 8). People also feel that extra care is needed for children and people outside the household—they want protection *for* them but equally *from* them. (These findings are largely consistent with much other work on privacy [168].) As these are the top-of-mind concerns, developers of always-listening services may wish to prioritize addressing them: having appropriate guarantees will assuage many of the most common fears.

Beyond these, many people do not consider private conversations themselves to be, de facto, sensitive; what matters is how they are used (Chapters 4, 7). Here, contextual integrity (Chapter 3) proves to be a good model for passive listening. As it would predict, there is no blanket sensitivity assigned to in-home data flows, and most people are fine with some of their speech being shared. But people do care a lot about the flows being appropriate; for example, they considered data being sold or shared with advertisers to be unacceptable. Ultimately, people want to know what is happening with their data. Nobody wants to be surprised, like what happened when consumers learned that humans were reviewing voice assistant recordings—these occurrences are norm violations and can destroy trust in a product and a company.

Privacy protection approaches

Another area this dissertation explored is the design space for potential privacy-enhancing technologies for always-listening devices. Chapter 5 contributes a taxonomy of privacy-preserving techniques for passive listening, as well as a research methodology for how to evaluate them. In Chapter 6, we demonstrated the viability of these methods, and began collecting data on them, with a study on user perceptions of these approaches. Our results can be used, for example, to identify the most promising directions to pursue.

Chapters 6 and 7 investigated specific privacy controls in greater detail. We found that

both install-time and runtime permissions were welcomed by people for the control that they provided, but that their user experience left more to be desired. Chapter 8 studied another approach—transparency—finding it to be a potentially promising avenue for building user trust, as well as catching misbehaving apps.

Methodology

If passive listening assistants, as we envisioned them here, are not realized exactly, there will certainly be other always-listening devices—we can say so with confidence, because they already exist. Today's assistants will get smarter and more ambitious, and more Internet of Things devices will appear that are always listening or continuously sensing and analyzing our data. When they do, their success and acceptance might depend not only on their functionality but on how well they are able to respect their users' privacy. In addition to the specific concerns and suggestions from this dissertation, its techniques may be useful for researchers who are thinking about how to study other devices that do not currently exist on the market.

We found that it is possible, if challenging, to study devices that do not exist yet. One thing we needed to do was explain them to our participants. This went fairly well: in all of our studies, people caught on quickly, answered comprehension check questions correctly, and provided responses relevant to our research questions, showing that they understood our vision for these devices. A methodological drawback that we observed was that explaining the hypothetical devices and their operation took a significant chunk of time in all of our studies—cutting into the subjects' limited time and attention.

We also found that, if a device does not exist, it is possible to simulate parts of it and get authentic feedback. Two key methodological insights contributed to this. First, in Chapters 7 and 8, we were able to use interactive real-time browser-based technologies to convincingly simulate Artificial Intelligence for our participants, yielding genuine reactions and realistic feedback. While the Wizard of Oz technique, which we used for this purpose, is not new, it is not as common in privacy research as it is in other subfields of human-computer interaction.

The second methodological choice made in this dissertation was to study the various components and user interfaces of a passive listening devices separately—for example, the installation process, runtime feedback, and auditing and review capabilities—as opposed to evaluating a single artifact as a whole. While the latter method would yield greater ecological validity, it would also introduce numerous confounds. In contrast, our approach allowed for cleaner consideration and comparison of potential variables. We therefore find it to be a compelling technique for studying hypothetical technologies, for which the design decisions have not been made yet, and therefore there are different ways the final product may turn out.

9.2 Open questions

The research in this dissertation begins the work to understand privacy for always-listening devices, but there is much more to explore. Here are some areas that could use more attention.

Survey responses and conversations with our participants underscore the importance of privacy education. People are far from ignorant about privacy, but there are pervasive misconceptions, both about the behavior of devices and the legal protections available to consumers. These have the potential to negatively affect people's decision-making. Products themselves do not do a good job educating users about their privacy options and consequences. Other information sources, such as the media, may be intermittent, incomplete, and biased. It is more important than ever for people to know what happens with their data and what *could* happen with their data. What is the best time to teach this and how?

A major source of misunderstandings is that users are often unaware of the data flows emanating from their devices. The opaqueness may well be deliberate, but it is also true that conveying this information is difficult. We need better transparency mechanisms, ideally ones that can be incorporated directly into products. People want to know what is happening with their data, but finding this out is too hard, and it is rarely a primary task for users, so people forget or lack time for it. How can transparency techniques be more effective?

Transparency inevitably necessitates data collection, so that the relevant information can be visualized for the user. But holding on to this data creates privacy risks due to the potential for it to be exposed or leaking bits of information. This is a general problem, but it is especially acute for always-listening devices, due to the volume and sensitivity of the data that passes through their microphones and memory modules. What is the best way to balance data retention for auditing and transparency with the need to minimize data for privacy?

As we have seen throughout this dissertation, people want choices, often demanding fine-grained control over what happens with their data. Yet research across disciplines has shown that, when there are too many options, people end up not taking advantage of them. Privacy settings hidden behind confusing menu options have become a well-known deceptive pattern. But forcing people to choose before they are ready is also ineffective (as we have seen in our work). How should we balance user choice with user engagement?

An added complication that is specific to always-listening devices is that their interfaces are likely to be voice-based. In this dissertation, we largely assumed that a screen would also be available, but this is unlikely to always be true, and the screen may, in any case, be

inconvenient. What is the best user experience when controls are complicated and require lots of information, but the device is primarily audio-based? How can the privacy choices be conveyed over audio?

Our research has shown that, unsurprisingly, there is a reasonably high degree of heterogeneity between people's privacy preferences. A one-size-fits-all solution is therefore unlikely to satisfy everyone. How can privacy choices be individualized?

The challenge of individualization for smart speakers and other always-listening devices is that they are often shared between members of a household, who may turn out to have different opinions about how their privacy should be managed. This is further complicated by potential intra-household relationship dynamics and power imbalances. How can heterogeneous privacy preferences be resolved?

The last, most basic, observation is that none of the approaches we explored so far have proven to be ideal. Are there better options for privacy controls for always-listening devices?

Legal and ethical considerations

Always-listening devices raise a host of legal and ethical issues. This dissertation has not aimed to explore these exhaustively (nor resolve them), and they remain open questions for other scholars to address.

GDPR and other privacy laws The European General Data Protection Regulation (GDPR), and newer privacy laws in several US states, such as California [39], Virginia [214], and Colorado [53], contain a number of provisions that may affect the deployment of always-listening devices. How do requirements like data minimization and limitations on storage interact with the smart speaker's need to hear and analyze all audio around it? Can the device be transparent about what it heard without creating a conversation log that may run afoul of further regulations (and may make many users uncomfortable)? What additional liabilities are added by other privacy rights, like the "Right to Be Forgotten"?

Wiretap laws The United States currently lacks comprehensive federal privacy legislation like Europe's GDPR (though some narrowly targeted laws, like COPPA, the Children's Online Privacy Protection Act may still be applicable). However, in addition to the state-specific privacy laws mentioned above, a number of individual states have laws against wiretapping, requiring all parties to consent if their conversation is to be recorded. (Remaining states are single-party consent states, where only one party needs to provide consent.) How does an always-listening device collect consent?

Informed consent Whether under the US or European regulatory regime, informed consent is likely to be a core requirement for the deployment of always-listening devices. But

how should an assistant obtain consent? Current smart speakers sidestep these questions by requiring their owners to obtain consent from all parties who might be recorded. (They are typically located in the home, making this a more tractable task.) Will always-listening devices be able to get away with the same requirements, or will they need to be more proactive about obtaining consent?

Ethical concerns Always-listening devices are a potent force for surveillance, both within a household and on a larger scale, by companies and states. Having access to an individual's conversations (especially in a private place, like one's home) is ripe for abuse; the possibilities are, quite literally, Orwellian. Even outside any dystopian visions, a device that is privy to so many conversations is certain to exacerbate any power imbalances, both between people within the home and between consumers and the companies providing their services. Without appropriate checks and balances, they are likely to exacerbate any power imbalances. In light of all these concerns, it is reasonable to ask whether always-listening devices are even ethically desirable.

Regardless of one's stance on that particular question, always-listening devices are all but inevitable. As we have seen, in many ways, they are already here, even if law and ethics have not fully caught up. We therefore believe that robust regulations will be vital to preventing privacy violations.

9.3 Concluding remarks

Intelligent voice assistants are already convenient and will get immensely more helpful as their capabilities advance. Passively listening devices, if we do see them, may turn out to be fantastically useful— those currently on the market are already performing potentially life-saving tasks, like detecting fires or health issues. It is therefore quite possible that, in the future, living without them will be as unthinkable as life without a cell phone is today for many people.

This is what makes the need for better privacy controls so urgent. Instead of having the unwilling opt out and eventually give up or be left behind, it would be far better to find a way to offer the features while addressing the privacy concerns, by providing effective safeguards and guarantees.

The controls that protect our privacy can be technical, and that is what this dissertation has focused on. But they also can—and should—be societal. The technical means for the system of total control described by Orwell are already here, and will only get cheaper and easier. What is holding them back today (aside from some minor technical limitations) are largely the societal controls: the legal and ethical constraints that deem mass surveillance and other forms of data misuse unacceptable. We need to promote these values, and strengthen the institutions that support them, so that societal and technical

controls can compose and cooperate. If privacy, as the California constitution declares, is an inalienable right, then it definitely deserves a defense in depth.

References

[1] 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)*. USENIX Association, 2019. url: https://www.usenix.org/conference/soups2019/presentation/abdi.

[2] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y. Lim, and Mohan Kankanhalli. "Trends and Trajectories for Explainable, Accountable and Intelligible Systems: An HCI Research Agenda". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1–18. isbn: 978-1-4503-5620-6. doi: 10.1145/3173574.3174156. url: https://doi.org/10.1145/3173574.3174156.

[3] A. Adadi and M. Berrada. "Peeking inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)". In: *IEEE Access* 6 (2018), pp. 52138–52160. issn: 2169-3536. doi: 10.1109/ACCESS.2018.2870052.

[4] Rebecca Adaimi, Howard Yong, and Edison Thomaz. "Ok Google, What Am I Doing? Acoustic Activity Recognition Bounded by Conversational Assistant Interactions". In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5.1 (Mar. 2021). doi: 10.1145/3448090. url: https://doi.org/10.1145/3448090.

[5] Paige H. Adams and Craig H. Martell. "Topic Detection and Extraction in Chat". In: *2008 IEEE International Conference on Semantic Computing*. Aug. 2008, pp. 581–588. doi: 10.1109/ICSC.2008.61.

[6] Yuvraj Agarwal and Malcolm Hall. "ProtectMyPrivacy: Detecting and Mitigating Privacy Leaks on IOS Devices Using Crowdsourcing". In: *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*. MobiSys '13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 97–110. isbn: 978-1-4503-1672-9. doi: 10.1145/2462456.2464460. url: https://doi.org/10.1145/2462456.2464460.

[7] Imtiaz Ahmad, Rosta Farzan, Apu Kapadia, and Adam J. Lee. "Tangible Privacy: Towards User-Centric Sensor Designs for Bystander Privacy". In: *Proc. ACM Hum.-*

Comput. Interact. 4.CSCW2 (Oct. 2020). doi: 10.1145/3415187. url: https://doi.org/10.1145/3415187.

[8] Devdatta Akhawe and Adrienne Porter Felt. "Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness". In: *Proceedings of the 22nd USENIX Security Symposium*. 2013, pp. 257–272.

[9] Amazon. *Alexa Skills*. url: https://www.amazon.com/alexa-skills/b?ie=UTF8&node=13727921011.

[10] Amazon. *Amazon.Com Help: Alexa and Alexa Device FAQs*. url: https://www.amazon.com/gp/help/customer/display.html?nodeId=201602230.

[11] Amazon. *Require a Voice Code for Purchases with Alexa*. url: https://www.amazon.com/gp/help/customer/display.html?nodeId=GAA2RYUEDNT5ZSNK (visited on 08/01/2021).

[12] Amazon. *Skill Certification Requirements*. url: https://developer.amazon.com/en-US/docs/alexa/custom-skills/certification-requirements-for-custom-skills.html (visited on 08/01/2021).

[13] Amazon. *What Are Alexa Voice Profiles?* url: https://www.amazon.com/gp/help/customer/display.html?nodeId=GYCXKY2AB2QWZT2X (visited on 08/01/2021).

[14] Tawfiq Ammari, Jofish Kaye, Janice Y. Tsai, and Frank Bentley. "Music, Search, and IoT: How People (Really) Use Voice Assistants". In: *ACM Transactions on Computer-Human Interaction* 26.3 (Apr. 2019). issn: 1073-0516. doi: 10.1145/3311956. url: https://doi.org/10.1145/3311956.

[15] Salvatore Andolina, Valeria Orso, Hendrik Schneider, Khalil Klouche, Tuukka Ruotsalo, Luciano Gamberini, and Giulio Jacucci. "Investigating Proactive Search Support in Conversations". In: *Proceedings of the 2018 Designing Interactive Systems Conference*. DIS '18. ACM, 2018, pp. 1295–1307. isbn: 978-1-4503-5198-0. doi: 10.1145/3196709.3196734. url: http://doi.acm.org/10.1145/3196709.3196734.

[16] Julio Angulo and Martin Ortlieb. ""WTH..!?!" Experiences, Reactions, and Expectations Related to Online Privacy Panic Situations". In: *Eleventh Symposium on Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, July 2015, pp. 19–38. isbn: 978-1-931971-24-9. url: https://www.usenix.org/conference/soups2015/proceedings/presentation/angulo.

[17] Noah Apthorpe, Yan Shvartzshnaider, Arunesh Mathur, Dillon Reisman, and Nick Feamster. "Discovering Smart Home Internet of Things Privacy Norms Using Contextual Integrity". In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2.2 (July 2018), 59:1–59:23. issn: 2474-9567. doi: 10.1145/3214262. url: http://doi.acm.org/10.1145/3214262.

[18] Orlando Arias, Jacob Wurm, Khoa Hoang, and Yier Jin. "Privacy and Security in Internet of Things and Wearable Devices". In: *IEEE Transactions on Multi-Scale Computing Systems* 1.2 (2015), pp. 99–109.

[19] Samuel Axon. "Amazon Halo Will Charge a Subscription Fee to Monitor the Tone of Your Voice". In: *Ars Technica* (Aug. 2020). url: https://arstechnica.com/gadgets/2020/08/amazon-halo-will-charge-a-subscription-fee-to-monitor-the-tone-of-your-voice/.

[20] Paritosh Bahirat, Martijn Willemsen, Yangyang He, Qizhang Sun, and Bart Knijnenburg. "Overlooking Context: How Do Defaults and Framing Reduce Deliberation in Smart Home Privacy Decision-Making?" In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8096-6. url: https://doi.org/10.1145/3411764.3445672.

[21] Lujo Bauer, Lorrie Faith Cranor, Saranga Komanduri, Michelle L Mazurek, Michael K Reiter, Manya Sleeper, and Blase Ur. "The Post Anachronism: The Temporal Dimension of Facebook Privacy". In: *Proceedings of the 12th ACM Workshop on Privacy in the Electronic Society*. ACM. 2013, pp. 1–12.

[22] Jeff Beer. "Facebook Says Sorry (Sort Of) In Its Biggest Ever Ad Campaign". In: *Fast Company* (Apr. 2018). url: https://www.fastcompany.com/40563382/facebook-says-sorry-sort-of-in-its-biggest-ever-ad-campaign.

[23] Sebastian Benthall, Seda Gürses, and Helen Nissenbaum. "Contextual Integrity through the Lens of Computer Science". In: *Foundations and Trends® in Privacy and Security* 2.1 (2017), pp. 1–69. issn: 2474-1558. doi: 10.1561/3300000016. url: http://dx.doi.org/10.1561/3300000016.

[24] Frank Bentley, Chris Luvogt, Max Silverman, Rushani Wirasinghe, Brooke White, and Danielle Lottridge. "Understanding the Long-Term Use of Smart Speaker Assistants". In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2.3 (Sept. 2018), 91:1–91:24. issn: 2474-9567. doi: 10.1145/3264901. url: http://doi.acm.org/10.1145/3264901.

[25] Julia Bernd, Ruba Abu-Salma, and Alisa Frik. "Bystanders' Privacy: The Perspectives of Nannies on Smart Home Surveillance". In: *10th USENIX Workshop on Free and Open Communications on the Internet (FOCI 20)*. USENIX Association, Aug. 2020. url: https://www.usenix.org/conference/foci20/presentation/bernd.

[26] Umang Bhatt et al. "Explainable Machine Learning in Deployment". In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. FAT* '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 648–657.

isbn: 978-1-4503-6936-7. doi: 10.1145/3351095.3375624. url: https://doi.org/10.1145/3351095.3375624.

[27] Asia J. Biega, Peter Potash, Hal Daumé, Fernando Diaz, and Michèle Finck. "Operationalizing the Legal Principle of Data Minimization for Personalization". In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 399–408. isbn: 978-1-4503-8016-4. url: https://doi.org/10.1145/3397271.3401034.

[28] Sushil Bikhchandani, David Hirshleifer, and Ivo Welch. "Learning from the Behavior of Others: Conformity, Fads, and Informational Cascades". In: *Journal of Economic Perspectives* 12.3 (Sept. 1998), pp. 151–170. doi: 10.1257/jep.12.3.151. url: https://www.aeaweb.org/articles?id=10.1257/jep.12.3.151.

[29] Matt Bishop, Elisabeth Sullivan, and Michelle Ruppel. *Computer Security: Art and Science*. Second edition. Boston: Addison-Wesley, 2019. isbn: 978-0-321-71233-2.

[30] Karl Bode. "Your Smart Electricity Meter Can Easily Spy On You, Court Ruling Warns". In: *Motherboard* (Aug. 2018). url: https://www.vice.com/en/article/j5n3pb/your-smart-electricity-meter-can-easily-spy-on-you-court-ruling-warns.

[31] Dieter Bohn. "Amazon Says 100 Million Alexa Devices Have Been Sold". In: *The Verge* (Jan. 2019). url: https://www.theverge.com/2019/1/4/18168565/amazon-alexa-devices-how-many-sold-number-100-million-dave-limp.

[32] Brennen Bouwmeester, Elsa Rodríguez, Carlos Gañán, Michel van Eeten, and Simon Parkin. ""The Thing Doesn't Have a Name": Learning from Emergent Real-World Interventions in Smart Home Security". In: *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*. USENIX Association, Aug. 2021, pp. 493–512. isbn: 978-1-939133-25-0. url: https://www.usenix.org/conference/soups2021/presentation/bouwmeester.

[33] Phoebe Braithwaite. "Smart Home Tech Is Being Turned into a Tool for Domestic Abuse". In: *Wired* (July 2018). url: https://www.wired.co.uk/article/internet-of-things-smart-home-domestic-abuse.

[34] Virginia Braun and Victoria Clarke. "Using Thematic Analysis in Psychology". en. In: *Qualitative Research in Psychology* 3.2 (Jan. 2006), pp. 77–101. issn: 1478-0887, 1478-0895. doi: 10.1191/1478088706qp063oa. url: http://www.tandfonline.com/doi/abs/10.1191/1478088706qp063oa (visited on 08/04/2021).

[35] Barry Brown, Moira McGregor, and Donald McMillan. "Searchable Objects: Search in Everyday Conversation". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW '15. New York, NY, USA: Association for Computing Machinery, 2015, pp. 508–517. isbn: 978-1-4503-2922-4. doi: 10.1145/2675133.2675206. url: https://doi.org/10.1145/2675133.2675206.

[36] Rich Brown and Molly Price. "Speak, Shower and Shave: Kohler Brings Smarts to Your Bathroom". In: *CNET News* (Jan. 2018). url: https://www.cnet.com/news/speak-shower-and-shave-kohler-brings-smarts-to-your-bathroom/.

[37] J. Bugeja, A. Jacobsson, and P. Davidsson. "On Privacy and Security Challenges in Smart Connected Homes". In: *2016 European Intelligence and Security Informatics Conference (EISIC)*. Aug. 2016, pp. 172–175. doi: 10.1109/EISIC.2016.044.

[38] Carrie J. Cai, Jonas Jongejan, and Jess Holbrook. "The Effects of Example-Based Explanations in a Machine Learning Interface". In: *Proceedings of the 24th International Conference on Intelligent User Interfaces*. IUI '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 258–262. isbn: 978-1-4503-6272-6. doi: 10.1145/3301275.3302289. url: https://doi.org/10.1145/3301275.3302289.

[39] *California Consumer Privacy Act*. 2018. url: https://www.oag.ca.gov/privacy/ccpa.

[40] Aylin Caliskan Islam, Jonathan Walsh, and Rachel Greenstadt. "Privacy Detective: Detecting Private Information and Collective Privacy Behavior in a Large Social Network". In: *Proceedings of the 13th Workshop on Privacy in the Electronic Society*. WPES '14. New York, NY, USA: Association for Computing Machinery, 2014, pp. 35–46. isbn: 978-1-4503-3148-7. doi: 10.1145/2665943.2665958. url: https://doi.org/10.1145/2665943.2665958.

[41] Weicheng Cao, Chunqiu Xia, Sai Teja Peddinti, David Lie, Nina Taft, and Lisa M. Austin. "A Large Scale Study of User Behavior, Expectations and Engagement with Android Permissions". In: *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 803–820. isbn: 978-1-939133-24-3. url: https://www.usenix.org/conference/usenixsecurity21/presentation/cao-weicheng.

[42] Juan Pablo Carrascal, Rodrigo De Oliveira, and Mauro Cherubini. "To Call or to Recall? That's the Research Question". In: *ACM Transactions on Computer-Human Interaction* 22.1 (Mar. 2015). issn: 1073-0516. doi: 10.1145/2656211. url: https://doi.org/10.1145/2656211.

[43] George Chalhoub, Martin J Kraemer, Norbert Nthala, and Ivan Flechais. ""It Did Not Give Me an Option to Decline": A Longitudinal Analysis of the User Experience of Security and Privacy in Smart Home Products". In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8096-6. doi: 10.1145/3411764.3445691. url: https://doi.org/10.1145/3411764.3445691.

[44] Varun Chandrasekaran, Suman Banerjee, Bilge Mutlu, and Kassem Fawaz. "PowerCut and Obfuscator: An Exploration of the Design Space for Privacy-Preserving Interventions for Smart Speakers". In: *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*. USENIX Association, Aug. 2021, pp. 535–552. isbn: 978-1-939133-25-0. url: https://www.usenix.org/conference/soups2021/presentation/chandrasekaran.

[45] Varun Chandrasekaran, Kassem Fawaz, Bilge Mutlu, and Suman Banerjee. "Characterizing Privacy Perceptions of Voice Assistants: A Technology Probe Study". In: *arXiv preprint arXiv:1812.00263* (2018). arXiv: 1812.00263.

[46] Chatham House. *Chatham House Rule*. url: https://web.archive.org/web/20210715205301/https://www.chathamhouse.org/about-us/chatham-house-rule (visited on 07/15/2021).

[47] Yuxin Chen, Huiying Li, Shan-Yuan Teng, Steven Nagels, Zhijing Li, Pedro Lopes, Ben Y. Zhao, and Haitao Zheng. "Wearable Microphone Jamming". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–12. isbn: 978-1-4503-6708-0. url: https://doi.org/10.1145/3313831.3376304.

[48] Long Cheng, Christin Wilson, Song Liao, Jeffrey Young, Daniel Dong, and Hongxin Hu. "Dangerous Skills Got Certified: Measuring the Trustworthiness of Amazon Alexa Platform". In: *ACM Conference on Computer and Communications Security (CCS)*. 2020. url: https://www.ftc.gov/system/files/documents/public_events/1548288/privacycon-2020-christin_wilson.pdf.

[49] Eugene Cho. "Hey Google, Can I Ask You Something in Private?" In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. ACM, 2019, 258:1–258:9. isbn: 978-1-4503-5970-2. doi: 10.1145/3290605.3300488. url: http://doi.acm.org/10.1145/3290605.3300488.

[50] Catalin Cimpanu. "Academics Smuggle 234 Policy-Violating Skills on the Alexa Skills Store". In: *ZDNet* (July 2020). url: https://www.zdnet.com/article/academics-smuggle-234-policy-violating-skills-on-the-alexa-skills-store/.

[51] Catalin Cimpanu. "Alexa and Google Home Devices Leveraged to Phish and Eavesdrop on Users, Again". In: *ZDNet* (Oct. 2019). url: https://www.zdnet.com/article/alexa-and-google-home-devices-leveraged-to-phish-and-eavesdrop-on-users-again/.

[52] Jessica Colnago, Yuanyuan Feng, Tharangini Palanivel, Sarah Pearman, Megan Ung, Alessandro Acquisti, Lorrie Faith Cranor, and Norman Sadeh. "Informing the Design of a Personalized Privacy Assistant for the Internet of Things". en. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. Honolulu HI USA: ACM, Apr. 2020, pp. 1–13. isbn: 978-1-4503-6708-0. doi: 10.1145/3313831.3376389. url: https://dl.acm.org/doi/10.1145/3313831.3376389 (visited on 06/25/2020).

[53] *Colorado Privacy Act*. 2021. url: https://leg.colorado.gov/bills/sb21-190.

[54] Committee on Health Research and the Privacy of Health Information: The HIPAA Privacy Rule, Board on Health Sciences Policy, Board on Health Care Services, and Institute of Medicine. *Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research*. Ed. by Sharyl J. Nass, Laura A. Levit, and Lawrence O. Gostin. Washington, D.C.: National Academies Press, Feb. 2009, p. 12458. isbn: 978-0-309-12499-7. doi: 10.17226/12458. url: http://www.nap.edu/catalog/12458.

[55] Joseph Cox. "Revealed: Microsoft Contractors Are Listening to Some Skype Calls". In: *Motherboard* (Aug. 2019). url: https://www.vice.com/en/article/xweqbq/microsoft-contractors-listen-to-skype-calls.

[56] Lorrie Faith Cranor and Simson Garfinkel, eds. *Security and Usability: Designing Secure Systems That People Can Use*. Beijing ; Sebastapol, CA: O'Reilly, 2005. isbn: 978-0-596-00827-7.

[57] Bart Custers, Francien Dechesne, Wolter Pieters, Bart Willem Schermer, and Simone van der Hof. "Consent and Privacy". In: *The Routledge Handbook of the Ethics of Consent*. Routledge, 2018, pp. 247–258. url: https://ssrn.com/abstract=3383465.

[58] Paul Cutsinger. *How to Improve Alexa Skill Discovery with Name-Free Interaction and More*. Sept. 2018. url: https://developer.amazon.com/blogs/alexa/post/0fecdb38-97c9-48ac-953b-23814a469cfc/skill-discovery.

[59] Nils Dahlbäck, Arne Jönsson, and Lars Ahrenberg. "Wizard of Oz Studies: Why and How". In: *Proceedings of the 1st International Conference on Intelligent User Interfaces*. IUI '93. New York, NY, USA: Association for Computing Machinery, 1993, pp. 193–200. isbn: 0-89791-556-9. doi: 10.1145/169891.169968. url: https://doi.org/10.1145/169891.169968.

[60] Anupam Das, Martin Degeling, Daniel Smullen, and Norman Sadeh. "Personalized Privacy Assistants for the Internet of Things: Providing Users with Notice and Choice". In: *IEEE Pervasive Computing* 17.3 (July 2018), pp. 35–46. issn: 1536-1268. doi: 10.1109/MPRV.2018.03367733. url: https://doi.org/10.1109/MPRV.2018.03367733.

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

[62] Dorothy Elizabeth Robling Denning. *Cryptography and Data Security*. Reading, Mass: Addison-Wesley, 1982. isbn: 978-0-201-10150-8.

[63] Tamara Denning, Cynthia Matuszek, Karl Koscher, Joshua R. Smith, and Tadayoshi Kohno. "A Spotlight on Security and Privacy Risks with Future Household Robots: Attacks and Lessons". In: *Proceedings of the 11th International Conference on Ubiquitous Computing*. UbiComp '09. ACM, 2009, pp. 105–114. isbn: 978-1-60558-431-7. doi: 10.1145/1620545.1620564. url: http://doi.acm.org/10.1145/1620545.1620564.

[64] Peter Dockrill. "Newly Released Amazon Patent Shows Just How Much Creepier Alexa Can Get". In: *Science Alert* (May 2019). url: https://www.sciencealert.com/creepy-new-amazon-patent-would-mean-alexa-records-everything-you-say-from-now-on.

[65] Daniel J. Dubois, Roman Kolcun, Anna Maria Mandalari, Muhammad Talha Paracha, David Choffnes, and Hamed Haddadi. "When Speakers Are All Ears: Characterizing Misactivations of IoT Smart Speakers". In: *Proceedings on Privacy Enhancing Technologies* 2020.4 (Oct. 2020), pp. 255–276. issn: 2299-0984. doi: 10.2478/popets-2020-0072. url: https://content.sciendo.com/view/journals/popets/2020/4/article-p255.xml (visited on 08/28/2020).

[66] Lisa Eadicicco. "How to Get Amazon's Alexa to Delete Everything You've Said to Your Echo Just by Asking". In: *Business Insider* (May 2019). url: https://www.businessinsider.com/amazon-alexa-can-delete-everything-you-say-how-to-ask-2019-5.

[67] Timothy H. Edgar. *Beyond Snowden: Privacy, Mass Surveillance, and the Struggle to Reform the NSA*. Washington, D.C: Brookings Institution Press, 2017. isbn: 978-0-8157-3063-7.

[68] Serge Egelman and Eyal Peer. "Predicting Privacy and Security Attitudes". In: *ACM SIGCAS Comput. Soc.* 45.1 (Feb. 2015), pp. 22–28. issn: 0095-2737. doi: 10.1145/2738210.2738215. url: https://doi.org/10.1145/2738210.2738215.

[69] Malin Eiband, Sarah Theres Völkel, Daniel Buschek, Sophia Cook, and Heinrich Hussmann. "When People and Algorithms Meet: User-Reported Problems in Intelligent Everyday Applications". In: *Proceedings of the 24th International Conference on Intelligent User Interfaces*. IUI '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 96–106. isbn: 978-1-4503-6272-6. doi: 10.1145/3301275.3302262. url: https://doi.org/10.1145/3301275.3302262.

[70] Yusra Elbitar, Michael Schilling, Trung Tin Nguyen, Michael Backes, and Sven Bugiel. "Explanation Beats Context: The Effect of Timing & Rationales on Users' Runtime Permission Decisions". In: *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 785–802. isbn: 978-1-939133-24-3. url: https://www.usenix.org/conference/usenixsecurity21/presentation/elbitar.

[71] European Parliament and the Council of the European Union. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC*. 2016. url: http://data.europa.eu/eli/reg/2016/679/oj.

[72] Sascha Fahl, Marian Harbach, Yasemin Acar, and Matthew Smith. "On the Ecological Validity of a Password Study". In: *Proceedings of the Ninth Symposium on Usable Privacy and Security*. SOUPS '13. Newcastle, United Kingdom: ACM, 2013, 13:1–13:13. isbn: 978-1-4503-2319-2. doi: 10.1145/2501604.2501617. url: http://doi.acm.org/10.1145/2501604.2501617.

[73] Brown Farinholt, Mohammad Rezaeirad, Paul Pearce, Hitesh Dharmdasani, Haikuo Yin, Stevens Le Blond, Damon McCoy, and Kirill Levchenko. "To Catch a Ratter: Monitoring the Behavior of Amateur DarkComet RAT Operators in the Wild". In: *2017 IEEE Symposium on Security and Privacy (SP)*. May 2017, pp. 770–787. doi: 10.1109/SP.2017.48.

[74] Adrienne Porter Felt, Serge Egelman, Matthew Finifter, Devdatta Akhawe, and David Wagner. "How to Ask for Permission". In: *HotSec*. 2012.

[75] Adrienne Porter Felt, Serge Egelman, and David Wagner. "I've Got 99 Problems, but Vibration Ain't One: A Survey of Smartphone Users' Concerns". In: *Proceedings of the Second ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*. SPSM '12. Raleigh, North Carolina, USA: ACM, 2012, pp. 33–44. isbn: 978-1-4503-1666-8. doi: 10.1145/2381934.2381943. url: http://doi.acm.org/10.1145/2381934.2381943.

[76] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, and David Wagner. "Android Permissions: User Attention, Comprehension, and Behavior". In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS '12. New York, NY, USA: ACM, 2012, 3:1–3:14. isbn: 978-1-4503-1532-6. doi:

10.1145/2335356.2335360. url: http://doi.acm.org/10.1145/2335356.2335360.

[77] Earlence Fernandes, Jaeyeon Jung, and Atul Prakash. "Security Analysis of Emerging Smart Home Applications". In: *2016 IEEE Symposium on Security and Privacy (SP)*. May 2016, pp. 636–654. doi: 10.1109/SP.2016.44.

[78] Matthew Finifter, Devdatta Akhawe, and David Wagner. "An Empirical Study of Vulnerability Rewards Programs". In: *22nd USENIX Security Symposium (USENIX Security 13)*. Washington, D.C.: USENIX Association, Aug. 2013, pp. 273–288. isbn: 978-1-931971-03-4. url: https://www.usenix.org/conference/usenixsecurity13/technical-sessions/presentation/finifter.

[79] Robyn Fisher. *Alexa Skills Challenge Offers $250,000 in Prizes for Best Kid Skills*. Oct. 2017. url: https://developer.amazon.com/blogs/alexa/post/f9671946-9039-45a4-83a7-ed1e15de682d/alexa-skills-challenge-offers-250-000-in-prizes-for-best-kid-skills.

[80] Geoffrey A. Fowler. "Alexa Has Been Eavesdropping on You This Whole Time". In: *The Washington Post* (May 2019). url: https://www.washingtonpost.com/technology/2019/05/06/alexa-has-been-eavesdropping-you-this-whole-time/.

[81] Sarah Frier. "Facebook Paid Contractors to Transcribe Users' Audio Chats". In: *Bloomberg* (Aug. 2019). url: https://www.bloomberg.com/news/articles/2019-08-13/facebook-paid-hundreds-of-contractors-to-transcribe-users-audio.

[82] Alisa Frik, Julia Bernd, Noura Alomar, and Serge Egelman. "A Qualitative Model of Older Adults' Contextual Decision-Making about Information Sharing". In: *Workshop on the Economics of Information Security (WEIS 2020)*. 2020.

[83] Jianfeng Gao, Michel Galley, and Lihong Li. *Neural Approaches to Conversational AI: Question Answering, Task-Oriented Dialogues and Social Chatbots*. 2019. url: https://ieeexplore.ieee.org/document/8649787.

[84] Xianyi Gao, Yulong Yang, Huiqing Fu, Janne Lindqvist, and Yang Wang. "Private Browsing: An Inquiry on Usability and Privacy Protection". In: *Proceedings of the 13th Workshop on Privacy in the Electronic Society*. WPES '14. New York, NY, USA: Association for Computing Machinery, 2014, pp. 97–106. isbn: 978-1-4503-3148-7. doi: 10.1145/2665943.2665953. url: https://doi.org/10.1145/2665943.2665953.

[85] Gartner, Inc. *Gartner Says Worldwide Spending on VPA-Enabled Wireless Speakers Will Top $3.5 Billion by 2021*. Tech. rep. Stamford, Connecticut, Aug. 2017. url: https://www.gartner.com/en/newsroom/press-releases/2017-08-24-gartner-says-worldwide-spending-on-vpa-enabled-wireless-speakers-will-top-3-billion-by-2021.

[86] Christine Geeng and Franziska Roesner. "Who's In Control?: Interactions In Multi-User Smart Homes". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. ACM, 2019, 268:1–268:13. isbn: 978-1-4503-5970-2. doi: 10.1145/3290605.3300498. url: http://doi.acm.org/10.1145/3290605.3300498.

[87] Makda Ghebreslassie. "'Stalked within Your Own Home': Woman Says Abusive Ex Used Smart Home Technology against Her". In: *CBC News* (Nov. 2018). url: https://www.cbc.ca/news/technology/tech-abuse-domestic-abuse-technology-marketplace-1.4864443.

[88] Arup Kumar Ghosh, Karla Badillo-Urquiola, Mary Beth Rosson, Heng Xu, John M. Carroll, and Pamela J. Wisniewski. "A Matter of Control or Safety? Examining Parental Use of Technical Monitoring Apps on Teens' Mobile Devices". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2018, pp. 1–14. isbn: 978-1-4503-5620-6. url: https://doi.org/10.1145/3173574.3173768.

[89] Elizabeth Goodman, Mike Kuniavsky, and Andrea Moed. *Observing the User Experience: A Practitioner's Guide to User Research*. 2nd ed. Amsterdam ; Boston: Morgan Kaufmann, 2012. isbn: 978-0-12-384869-7.

[90] Google. *Android 6.0 Changes*. 2015. url: https://developer.android.com/about/versions/marshmallow/android-6.0-changes.

[91] Google. *Data Security & Privacy on Google Home*. url: https://support.google.com/googlehome/answer/7072285?hl=en.

[92] Google. *Link Your Voice to Your Devices with Voice Match*. url: https://support.google.com/assistant/answer/9071681 (visited on 08/01/2021).

[93] Jorge Granjal, Edmundo Monteiro, and Jorge Sá Silva. "Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues". In: *IEEE Communications Surveys & Tutorials* 17.3 (2015), pp. 1294–1312.

[94] Jay Greene. "Amazon Adds Delete Commands for Alexa". In: *The Washington Post* (May 2019). url: https://www.washingtonpost.com/technology/2019/05/29/amazon-adds-alexa-delete-commands/.

[95] Zhixiu Guo, Zijin Lin, Pan Li, and Kai Chen. "SkillExplorer: Understanding the Behavior of Skills in Large Scale". In: *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 2649–2666. isbn: 978-1-939133-17-5. url: https://www.usenix.org/conference/usenixsecurity20/presentation/guo.

[96] Hana Habib, Jessica Colnago, Vidya Gopalakrishnan, Sarah Pearman, Jeremy Thomas, Alessandro Acquisti, Nicolas Christin, and Lorrie Faith Cranor. "Away from Prying Eyes: Analyzing Usage and Understanding of Private Browsing". In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 159–175. isbn: 978-1-939133-10-6. url: https://www.usenix.org/conference/soups2018/presentation/habib-prying.

[97] Lucy Handley. "Facebook Runs Ad Campaign That Sort of Says Sorry for Data Misuse Scandal". In: *CNBC* (Apr. 2018). url: https://www.cnbc.com/2018/04/26/facebook-runs-ad-campaign-somewhat-apologizing-for-data-misuse-scandal.html.

[98] Majid Hatamian, Jetzabel Serna, and Kai Rannenberg. "Revealing the Unrevealed: Mining Smartphone Users Privacy Perception on App Markets". In: *Computers & Security* 83 (2019), pp. 332–353. issn: 0167-4048. doi: 10.1016/j.cose.2019.02.010. url: http://www.sciencedirect.com/science/article/pii/S0167404818313051.

[99] W. He, J. Martinez, R. Padhi, L. Zhang, and B. Ur. "When Smart Devices Are Stupid: Negative Experiences Using Home Smart Devices". In: *2019 IEEE Security and Privacy Workshops (SPW)*. May 2019, pp. 150–155. doi: 10.1109/SPW.2019.00036.

[100] Abrar Al-Heeti. "Echo Effect: US Smart Speaker Ownership Nearly Doubles in a Year, Survey Says". In: *CNET News* (Feb. 2019). url: https://www.cnet.com/news/echo-effect-smart-speaker-ownership-nearly-doubles-in-a-year-survey-says/.

[101] Alex Hern. "Amazon's Halo Wristband: The Fitness Tracker That Listens to Your Mood". In: *The Guardian* (Aug. 2020). url: https://www.theguardian.com/technology/2020/aug/28/amazons-halo-wristband-the-fitness-tracker-that-listens-to-your-mood.

[102] Alex Hern. "Apple Contractors 'regularly Hear Confidential Details' on Siri Recordings". In: *The Guardian* (July 2019). url: https://www.theguardian.com/technology/2019/jul/26/apple-contractors-regularly-hear-confidential-details-on-siri-recordings.

[103] Christopher M. Hoadley, Heng Xu, Joey J. Lee, and Mary Beth Rosson. "Privacy as Information Access and Illusory Control: The Case of the Facebook News Feed Privacy Outcry". In: *Electronic Commerce Research and Applications* 9.1 (2010), pp. 50–60. issn: 1567-4223. doi: 10.1016/j.elerap.2009.05.001. url: https://www.sciencedirect.com/science/article/pii/S1567422309000271.

[104] Yue Huang, Borke Obada-Obieh, and Konstantin (Kosta) Beznosov. "Amazon vs. My Brother: How Users of Shared Smart Speakers Perceive and Cope with Privacy Risks". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–13. isbn: 978-1-4503-6708-0. doi: 10.1145/3313831.3376529. url: https://doi.org/10.1145/3313831.3376529.

[105] Yan Jia, Xingming Wang, Xiaoyi Qin, Yinping Zhang, Xuyang Wang, Junjie Wang, and Ming Li. "The 2020 Personalized Voice Trigger Challenge: Open Database, Evaluation Metrics and the Baseline Systems". In: *arXiv:2101.01935 [eess]* (Jan. 2021). arXiv: 2101.01935 [eess]. url: http://arxiv.org/abs/2101.01935.

[106] Jacob Kastrenakes. "LinkedIn Agrees to Settle Unwanted Email Lawsuit". In: *The Verge* (Oct. 2015). url: https://www.theverge.com/2015/10/2/9444067/linkedin-email-lawsuit-settlement-add-connections.

[107] J. F. Kelley. "An Empirical Methodology for Writing User-Friendly Natural Language Computer Applications". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '83. New York, NY, USA: Association for Computing Machinery, 1983, pp. 193–196. isbn: 0-89791-121-0. doi: 10.1145/800045.801609. url: https://doi.org/10.1145/800045.801609.

[108] Jason Kelley. *Students Are Pushing Back Against Proctoring Surveillance Apps*. Sept. 2020. url: https://www.eff.org/deeplinks/2020/09/students-are-pushing-back-against-proctoring-surveillance-apps.

[109] Sean Michael Kerner. "Researchers Find Amazon Alexa Can Be Hacked to Record Users". In: *eWeek* (Apr. 2018). url: https://www.eweek.com/security/researchers-find-amazon-alexa-can-be-hacked-to-record-users.

[110] Mohammad Taha Khan, Maria Hyun, Chris Kanich, and Blase Ur. "Forgotten But Not Gone: Identifying the Need for Longitudinal Data Management in Cloud Storage". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. New York, NY, USA: ACM, 2018, 543:1–543:12. isbn: 978-1-4503-5620-6. doi: 10.1145/3173574.3174117. url: http://doi.acm.org/10.1145/3173574.3174117.

[111] Jonathan Kilgour, Jean Carletta, and Steve Renals. "The Ambient Spotlight: Queryless Desktop Search from Meeting Speech". In: *Proceedings of the 2010 International Workshop on Searching Spontaneous Conversational Speech*. SSCS '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 49–52. isbn: 978-1-4503-0162-6. doi: 10.1145/1878101.1878112. url: https://doi.org/10.1145/1878101.1878112.

[112] Ilker Koksal. "The Sales Of Smart Speakers Skyrocketed". In: *Forbes* (Mar. 2020). url: https://www.forbes.com/sites/ilkerkoksal/2020/03/10/the-sales-of-smart-speakers-skyrocketed/.

[113] Saranga Komanduri, Richard Shay, Patrick Gage Kelley, Michelle L. Mazurek, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Serge Egelman. "Of Passwords and People: Measuring the Effect of Password-Composition Policies". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. Vancouver, BC, Canada: ACM, 2011, pp. 2595–2604. isbn: 978-1-4503-0228-9. doi: 10.1145/1978942.1979321. url: http://doi.acm.org/10.1145/1978942.1979321.

[114] Deguang Kong, Lei Cen, and Hongxia Jin. "AUTOREB: Automatically Understanding the Review-to-Behavior Fidelity in Android Applications". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. CCS '15. New York, NY, USA: Association for Computing Machinery, 2015, pp. 530–541. isbn: 978-1-4503-3832-5. doi: 10.1145/2810103.2813689. url: https://doi.org/10.1145/2810103.2813689.

[115] Vinay Koshy, Joon Sung Sung Park, Ti-Chung Cheng, and Karrie Karahalios. ""We Just Use What They Give Us": Understanding Passenger User Perspectives in Smart Homes". In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8096-6. url: https://doi.org/10.1145/3411764.3445598.

[116] Iga Kozlowska. *Facebook and Data Privacy in the Age of Cambridge Analytica*. Apr. 2018. url: https://jsis.washington.edu/news/facebook-data-privacy-age-cambridge-analytica/.

[117] Jacob Leon Kröger, Otto Hans-Martin Lutz, and Philip Raschke. "Privacy Implications of Voice and Speech Analysis – Information Disclosure by Inference". en. In: *Privacy and Identity Management. Data for Better Living: AI and Privacy*. Ed. by Michael Friedewald, Melek Önen, Eva Lievens, Stephan Krenn, and Samuel Fricker. Vol. 576. Cham: Springer International Publishing, 2020, pp. 242–258. isbn: 978-3-030-42503-6. doi: 10.1007/978-3-030-42504-3_16. url: http://link.springer.com/10.1007/978-3-030-42504-3_16.

[118] Deepak Kumar, Riccardo Paccagnella, Paul Murley, Eric Hennenfent, Joshua Mason, Adam Bates, and Michael Bailey. "Skill Squatting Attacks on Amazon Alexa". In: *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, 2018, pp. 33–47. isbn: 978-1-939133-04-5. url: https://www.usenix.org/conference/usenixsecurity18/presentation/kumar.

[119] Lawrence L. Kupper and Kerry B. Hafner. "On Assessing Interrater Agreement for Multiple Attribute Responses". In: *Biometrics* 45.3 (Sept. 1989), p. 957. issn: 0006341X. doi: 10.2307/2531695. url: http://www.jstor.org/stable/2531695.

[120] Christoffer Lambertsson. *Expectations of Privacy in Voice Interaction–A Look at Voice Controlled Bank Transactions*. Tech. rep. 2017.

[121] Reed Larson and Mihaly Csikszentmihalyi. "The Experience Sampling Method". In: *New Directions for Methodology of Social & Behavioral Science* 15 (1983), pp. 41–56. issn: 0271-1249.

[122] Josephine Lau, Benjamin Zimmerman, and Florian Schaub. "Alexa, Are You Listening?: Privacy Perceptions, Concerns and Privacy-Seeking Behaviors with Smart Speakers". In: *Proc. ACM Hum.-Comput. Interact.* 2.CSCW (Nov. 2018), 102:1–102:31. issn: 2573-0142. doi: 10.1145/3274371. url: http://doi.acm.org/10.1145/3274371.

[123] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research Methods in Human-Computer Interaction*. Chichester, West Sussex, U.K: Wiley, 2010. isbn: 978-0-470-72337-1.

[124] Linda Lee, JoongHwa Lee, Serge Egelman, and David Wagner. "Information Disclosure Concerns in The Age of Wearable Computing". In: *Proceedings of the NDSS Workshop on Usable Security*. USEC '16. 2016.

[125] Xingan Li. "A Review of Motivations of Illegal Cyber Activities". In: *Kriminologija & socijalna integracija* 25.1 (May 2017), pp. 110–126. issn: 18487963, 13302604. doi: 10.31299/ksi.25.1.4. url: https://hrcak.srce.hr/181161.

[126] Yao Li, Alfred Kobsa, Bart P. Knijnenburg, and M-H. Carolyn Nguyen. "Cross-Cultural Privacy Prediction". en. In: *Proceedings on Privacy Enhancing Technologies* 2017.2 (Apr. 2017), pp. 113–132. issn: 2299-0984. doi: 10.1515/popets-2017-0019. url: https://www.sciendo.com/article/10.1515/popets-2017-0019.

[127] Lily Hay Newman. "100 Million More IoT Devices Are Exposed—and They Won't Be the Last". In: *Wired* (Apr. 2021). url: https://www.wired.com/story/namewreck-iot-vulnerabilities-tcpip-millions-devices/.

[128] Lily Hay Newman. "An Alexa Bug Could Have Exposed Your Voice History to Hackers". In: *Wired* (Aug. 2020). url: https://www.wired.com/story/amazon-alexa-bug-exposed-voice-history-hackers/.

[129] Lily Hay Newman. "Turning an Echo Into a Spy Device Only Took Some Clever Coding". In: *Wired* (Apr. 2018). url: https://www.wired.com/story/amazon-echo-alexa-skill-spying/.

[130] Bin Liu, Mads Schaarup Andersen, Florian Schaub, Hazim Almuhimedi, Shikun (Aerin) Zhang, Norman Sadeh, Yuvraj Agarwal, and Alessandro Acquisti. "Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions". In: *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, June 2016, pp. 27–41. isbn: 978-1-931971-31-7. url: https://www.usenix.org/conference/soups2016/technical-sessions/presentation/liu.

[131] Bing Liu. "Sentiment Analysis and Opinion Mining". en. In: *Synthesis Lectures on Human Language Technologies* 5.1 (May 2012), pp. 1–167. issn: 1947-4040, 1947-4059. doi: 10.2200/S00416ED1V01Y201204HLT016. url: http://www.morganclaypool.com/doi/abs/10.2200/S00416ED1V01Y201204HLT016.

[132] Bing Liu and Ian Lane. "Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling". In: *Interspeech 2016*. Sept. 2016, pp. 685–689. doi: 10.21437/Interspeech.2016-1352. url: http://www.isca-speech.org/archive/Interspeech_2016/abstracts/1352.html.

[133] Yuchen Liu, Ziyu Xiang, Eun Ji Seong, Apu Kapadia, and Donald S. Williamson. "Defending Against Microphone-Based Attacks with Personalized Noise". en. In: *Proceedings on Privacy Enhancing Technologies* 2021.2 (Apr. 2021), pp. 130–150. issn: 2299-0984. doi: 10.2478/popets-2021-0021. url: https://petsymposium.org/2021/files/papers/issue2/popets-2021-0021.pdf.

[134] Kim Lyons. "Fitbit May Soon Be Adding Snoring Detection to Its Devices". In: *The Verge* (May 2021). url: https://www.theverge.com/2021/5/29/22459675/fitbit-snoring-detection-fitness-trackers-smartwatch-sleep-animal.

[135] Sapna Maheshwari. "Hey, Alexa, What Can You Hear? And What Will You Do With It?" In: *The New York Times* (Mar. 2018), A1. url: https://www.nytimes.com/2018/03/31/business/media/amazon-google-privacy-digital-assistants.html.

[136] Sapna Maheshwari. "Sharing Data for Deals? More Like Watching It Go With a Sigh". In: *The New York Times* (Dec. 2018), B1. url: https://www.nytimes.com/2018/12/24/business/media/data-sharing-deals-privacy.html.

[137] Thomas Maillart, Mingyi Zhao, Jens Grossklags, and John Chuang. "Given Enough Eyeballs, All Bugs Are Shallow? Revisiting Eric Raymond with Bug Bounty Programs". In: *Journal of Cybersecurity* 3.2 (Oct. 2017), pp. 81–90. issn: 2057-2085. doi: 10.1093/cybsec/tyx008. eprint: https://academic.oup.com/cybersecurity/article-pdf/3/2/81/23721621/tyx008.pdf. url: https://doi.org/10.1093/cybsec/tyx008.

[138] David J. Major, Danny Yuxing Huang, Marshini Chetty, and Nick Feamster. "Alexa, Who Am I Speaking To? Understanding Users' Ability to Identify Third-Party Apps on Amazon Alexa". In: *arXiv:1910.14112 [cs]* (Oct. 2019). arXiv: 1910.14112 [cs]. url: http://arxiv.org/abs/1910.14112 (visited on 05/19/2020).

[139] Nathan Malkin, Julia Bernd, Maritza Johnson, and Serge Egelman. ""What Can't Data Be Used For?": Privacy Expectations about Smart TVs in the U.S." en. In: *Proceedings 3rd European Workshop on Usable Security*. London, England: Internet Society, 2018. isbn: 978-1-891562-54-9. doi: 10.14722/eurousec.2018.23016. url: https://www.ndss-symposium.org/wp-content/uploads/2018/06/eurousec2018_16_Malkin_paper.pdf (visited on 12/14/2020).

[140] Lydia Manikonda, Aditya Deotale, and Subbarao Kambhampati. "What's Up with Privacy?: User Preferences and Privacy Concerns in Intelligent Personal Assistants". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. AIES '18. New York, NY, USA: ACM, 2018, pp. 229–235. isbn: 978-1-4503-6012-8. doi: 10.1145/3278721.3278773. url: http://doi.acm.org/10.1145/3278721.3278773.

[141] David Maulsby, Saul Greenberg, and Richard Mander. "Prototyping an Intelligent Agent through Wizard of Oz". In: *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*. CHI '93. New York, NY, USA: Association for Computing Machinery, 1993, pp. 277–284. isbn: 0-89791-575-5. doi: 10.1145/169059.169215. url: https://doi.org/10.1145/169059.169215.

[142] Moira McGregor and John C. Tang. "More to Meetings: Challenges in Using Speech-Based Technology to Support Meetings". In: *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17. New York, NY, USA: ACM, 2017, pp. 2208–2220. isbn: 978-1-4503-4335-0. doi: 10.1145/2998181.2998335. url: http://doi.acm.org/10.1145/2998181.2998335.

[143] Sheena McKenzie. "Facebook's Mark Zuckerberg Says Sorry in Full-Page Newspaper Ads". In: *CNN* (Mar. 2018). url: https://www.cnn.com/2018/03/25/europe/facebook-zuckerberg-cambridge-analytica-sorry-ads-newspapers-intl/index.html.

[144] Christopher Mele. "Bid for Access to Amazon Echo Audio in Murder Case Raises Privacy Concerns". In: *The New York Times* (Dec. 2016). url: https://www.nytimes.com/2016/12/28/business/amazon-echo-murder-case-arkansas.html (visited on 06/08/2018).

[145] Wei Meng, Ren Ding, Simon P. Chung, Steven Han, and Wenke Lee. "The Price of Free: Privacy Leakage in Personalized Mobile In-App Ads". en. In: *Proceedings 2016 Network and Distributed System Security Symposium*. San Diego, CA: Internet

Society, 2016. isbn: 978-1-891562-41-9. doi: 10.14722/ndss.2016.23353. url: https://www.ndss-symposium.org/wp-content/uploads/2017/09/price-of-free-privacy-leakage-personalized-mobile-in-app-ads.pdf (visited on 08/04/2021).

[146] Sarah Mennicken and Elaine M. Huang. "Hacking the Natural Habitat: An In-the-Wild Study of Smart Homes, Their Development, and the People Who Live in Them". In: *Pervasive Computing*. Ed. by Judy Kay, Paul Lukowicz, Hideyuki Tokuda, Patrick Olivier, and Antonio Krüger. Springer Berlin Heidelberg, 2012, pp. 143–160. isbn: 978-3-642-31205-2. doi: 10.1007/978-3-642-31205-2_10.

[147] Oliver Michler, Reinhold Decker, and Christian Stummer. "To Trust or Not to Trust Smart Consumer Products: A Literature Review of Trust-Building Factors". en. In: *Management Review Quarterly* 70.3 (Aug. 2020), pp. 391–420. issn: 2198-1620, 2198-1639. doi: 10.1007/s11301-019-00171-8. url: http://link.springer.com/10.1007/s11301-019-00171-8.

[148] Richard Mitev, Markus Miettinen, and Ahmad-Reza Sadeghi. "Alexa Lied to Me: Skill-Based Man-in-the-Middle Attacks on Virtual Assistants". In: *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*. Asia CCS '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 465–478. isbn: 978-1-4503-6752-3. doi: 10.1145/3321705.3329842. url: https://doi.org/10.1145/3321705.3329842.

[149] M. Mondal, J. Messias, S. Ghosh, K. Gummadi, and A. Kate. "Longitudinal Privacy Management in Social Media: The Need for Better Controls". In: *IEEE Internet Computing* (2018), pp. 1–1. issn: 1089-7801. doi: 10.1109/MIC.2017.265102818.

[150] David Monsees and Marlo McGriff. *Introducing Auto-Delete Controls for Your Location History and Activity Data*. May 2019. url: https://www.blog.google/technology/safety-security/automatically-delete-data/.

[151] Aarthi Easwara Moorthy. "Voice Activated Personal Assistant: Privacy Concerns in the Public Space". English. PhD Thesis. 2013. url: https://search.proquest.com/docview/1513579796?accountid=14496.

[152] Aarthi Easwara Moorthy and Kim-Phuong L. Vu. "Privacy Concerns for Use of Voice Activated Personal Assistant in the Public Space". In: *International Journal of Human–Computer Interaction* 31.4 (2015), pp. 307–335. doi: 10.1080/10447318.2014.986642. url: https://doi.org/10.1080/10447318.2014.986642.

[153] S. Murugesan, S. Malik, F. Du, E. Koh, and T. M. Lai. "DeepCompare: Visual and Interactive Comparison of Deep Learning Model Performance". In: *IEEE Computer Graphics and Applications* 39.5 (Sept. 2019), pp. 47–59. issn: 1558-1756. doi: 10.1109/MCG.2019.2919033.

[154] 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)*. USENIX Association, 2017, pp. 399–412. isbn: 978-1-931971-39-3. url: https://www.usenix.org/conference/soups2017/technical-sessions/presentation/naeini.

[155] Jared Newman. "You Can Now Buy an Amazon Echo Add-on That Stops Alexa from Listening". In: *Fast Company* (July 2020). url: https://www.fastcompany.com/90532150/you-can-now-buy-an-amazon-echo-add-on-that-stops-alexa-from-listening.

[156] D. C. Nguyen, E. Derr, M. Backes, and S. Bugiel. "Short Text, Large Effect: Measuring the Impact of User Reviews on Android App Security & Privacy". In: *2019 IEEE Symposium on Security and Privacy (SP)*. May 2019, pp. 555–569. doi: 10.1109/SP.2019.00012.

[157] Nick Nikiforakis, Alexandros Kapravelos, Wouter Joosen, Christopher Kruegel, Frank Piessens, and Giovanni Vigna. "Cookieless Monster: Exploring the Ecosystem of Web-Based Device Fingerprinting". In: *2013 IEEE Symposium on Security and Privacy*. May 2013, pp. 541–555. doi: 10.1109/SP.2013.43.

[158] Helen Nissenbaum. "Contextual Integrity Up and Down the Data Food Chain". In: *Theoretical Inquiries in Law* 20.1 (2019), pp. 221–256. doi: doi:10.1515/til-2019-0008. url: https://doi.org/10.1515/til-2019-0008.

[159] Helen Nissenbaum. "Privacy as Contextual Integrity". In: *Washington Law Review* 79 (Feb. 2004), p. 119.

[160] Helen Nissenbaum. *Privacy in Context: Technology, Policy, and the Integrity of Social Life*. Stanford, Calif: Stanford Law Books, 2009. isbn: 978-0-8047-5236-7.

[161] Marije Nouwen, Maarten Van Mechelen, and Bieke Zaman. "A Value Sensitive Design Approach to Parental Software for Young Children". In: *Proceedings of the 14th International Conference on Interaction Design and Children*. IDC '15. New York, NY, USA: Association for Computing Machinery, 2015, pp. 363–366. isbn: 978-1-4503-3590-4. doi: 10.1145/2771839.2771917. url: https://doi.org/10.1145/2771839.2771917.

[162] OECD. *Guidelines on the Protection of Privacy and Transborder Flows of Personal Data*. Tech. rep. Sept. 1980. url: https://www.oecd.org/digital/ieconomy/oecdguidelinesontheprotectionofprivacyandtransborderflowsofperson
htm#part2.

[163] Antti Oulasvirta, Aurora Pihlajamaa, Jukka Perkiö, Debarshi Ray, Taneli Vähäkangas, Tero Hasu, Niklas Vainio, and Petri Myllymäki. "Long-Term Effects of Ubiquitous Surveillance in the Home". In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. ACM. Pittsburgh, Pennsylvania, 2012, pp. 41–50.

[164] 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 (Oct. 2018), pp. 33–50. doi: 10.1515/popets-2018-0030. url: https://content.sciendo.com/view/journals/popets/2018/4/article-p33.xml.

[165] R. Parasuraman, T.B. Sheridan, and C.D. Wickens. "A Model for Types and Levels of Human Interaction with Automation". In: *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 30.3 (2000), pp. 286–297. doi: 10.1109/3468.844354.

[166] Katie Paul. "Facebook Separates Security Tool from Friend Suggestions, Citing Privacy Overhaul". In: *Reuters* (Dec. 2019). url: https://www.reuters.com/article/us-facebook-privacy/facebook-separates-security-tool-from-friend-suggestions-citing-privacy-overhaul-idUKKBN1YN26Q.

[167] Sarah Perez. "COVID-19 Quarantine Boosts Smart Speaker Usage among U.S. Adults, Particularly Younger Users". In: *Techcrunch* (Apr. 2020). url: https://techcrunch.com/2020/04/30/covid-19-quarantine-boosts-smart-speaker-usage-among-u-s-adults-particularly-younger-users/.

[168] Pew Research Center. *Public Perceptions of Privacy and Security in the Post-Snowden Era*. Tech. rep. Pew Research Center, Nov. 2014. url: https://www.pewinternet.org/2014/11/12/public-privacy-perceptions/https://www.pewinternet.org/2014/11/12/public-privacy-perceptions/.

[169] Phil Alden Robinson. *Sneakers*. 1992.

[170] Kurt Wesley Piersol and Gabriel Beddingfield. "Pre-Wakeword Speech Processing". US10643606B2. 2020.

[171] Alexander Ponticello, Matthias Fassl, and Katharina Krombholz. "Exploring Authentication for Security-Sensitive Tasks on Smart Home Voice Assistants". In: *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*. USENIX Association, Aug. 2021, pp. 475–492. isbn: 978-1-939133-25-0. url: https://www.usenix.org/conference/soups2021/presentation/ponticello.

[172] Ben Popper. "Google's New Clips Camera Is Invasive, Creepy, and Perfect for a Parent like Me". In: *The Verge* (Oct. 2017). url: https://www.theverge.com/2017/10/5/16428708/google-clips-camera-privacy-parents-children.

[173] Jon Porter. "Security Researchers Expose New Alexa and Google Home Vulnerability". In: *The Verge* (Oct. 2019). url: https://www.theverge.com/2019/10/21/20924886/alexa-google-home-security-vulnerability-srlabs-phishing-eavesdropping.

[174] J.M. Porup. ""Internet of Things" Security Is Hilariously Broken and Getting Worse". In: *Ars Technica* (Jan. 2016). url: https://arstechnica.com/information-technology/2016/01/how-to-search-the-internet-of-things-for-photos-of-sleeping-babies/.

[175] Abhinav Rastogi, Raghav Gupta, and Dilek Hakkani-Tur. "Multi-Task Learning for Joint Language Understanding and Dialogue State Tracking". In: *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*. Association for Computational Linguistics, July 2018, pp. 376–384. url: https://www.aclweb.org/anthology/W18-5045.

[176] Abbas Razaghpanah, Rishab Nithyanand, Narseo Vallina-Rodriguez, Srikanth Sundaresan, Mark Allman, Christian Kreibich, and Phillipa Gill. "Apps, Trackers, Privacy, and Regulators: A Global Study of the Mobile Tracking Ecosystem". en. In: *Proceedings 2018 Network and Distributed System Security Symposium*. San Diego, CA: Internet Society, 2018. isbn: 978-1-891562-49-5. doi: 10.14722/ndss.2018.23353. url: https://www.ndss-symposium.org/wp-content/uploads/2018/02/ndss2018_05B-3_Razaghpanah_paper.pdf.

[177] Robert W. Reeder, Adrienne Porter Felt, Sunny Consolvo, Nathan Malkin, Christopher Thompson, and Serge Egelman. "An Experience Sampling Study of User Reactions to Browser Warnings in the Field". In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. ACM, 2018, 512:1–512:13. isbn: 978-1-4503-5620-6. doi: 10.1145/3173574.3174086. url: http://doi.acm.org/10.1145/3173574.3174086.

[178] Irwin Reyes, Primal Wijesekera, Joel Reardon, Amit Elazari Bar On, Abbas Razaghpanah, Narseo Vallina-Rodriguez, and Serge Egelman. ""Won't Somebody Think of the Children?" Examining COPPA Compliance at Scale". In: *Proceedings on Privacy Enhancing Technologies* 2018.3 (June 2018), pp. 63–83. doi: 10.1515/popets-2018-0021. url: https://content.sciendo.com/view/journals/popets/2018/3/article-p63.xml.

[179] Laurel D. Riek. "Wizard of Oz Studies in HRI: A Systematic Review and New Reporting Guidelines". In: *J. Hum.-Robot Interact.* 1.1 (July 2012), pp. 119–136. doi: 10.5898/JHRI.1.1.Riek. url: https://doi.org/10.5898/JHRI.1.1.Riek.

[180] Eyal Ronen, Adi Shamir, Achi-Or Weingarten, and Colin O'Flynn. "IoT Goes Nuclear: Creating a ZigBee Chain Reaction". In: *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE. 2017, pp. 195–212.

[181] Sherif Saad, William Briguglio, and Haytham Elmiligi. "The Curious Case of Machine Learning In Malware Detection". In: *arXiv:1905.07573 [cs]* (May 2019). arXiv: 1905.07573 [cs]. url: http://arxiv.org/abs/1905.07573 (visited on 09/17/2020).

[182] Katharine Sarikakis and Lisa Winter. "Social Media Users' Legal Consciousness about Privacy". In: *Social Media + Society* 3.1 (2017), p. 2056305117695325. doi: 10.1177/2056305117695325. eprint: https://doi.org/10.1177/2056305117695325. url: https://doi.org/10.1177/2056305117695325.

[183] Ruhi Sarikaya. *Making Alexa More Friction-Free*. Apr. 2018. url: https://developer.amazon.com/blogs/alexa/post/60e1f011-3236-4162-b0f6-509205d354ca/making-alexa-more-friction-free.

[184] Mark Savage. "Spotify Wants to Suggest Songs Based on Your Emotions". In: *BBC News* (Jan. 2021). url: https://www.bbc.com/news/entertainment-arts-55839655.

[185] Florian Schaub, Rebecca Balebako, Adam L. Durity, and Lorrie Faith Cranor. "A Design Space for Effective Privacy Notices". In: *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, 2015, pp. 1–17. isbn: 978-1-931971-24-9. url: https://www.usenix.org/conference/soups2015/proceedings/presentation/schaub.

[186] Lea Schönherr, Maximilian Golla, Thorsten Eisenhofer, Jan Wiele, Dorothea Kolossa, and Thorsten Holz. "Unacceptable, Where Is My Privacy? Exploring Accidental Triggers of Smart Speakers". In: *arXiv:2008.00508 [cs]* (Aug. 2020). arXiv: 2008.00508 [cs]. url: http://arxiv.org/abs/2008.00508.

[187] John S. Seberger, Marissel Llavore, Nicholas Nye Wyant, Irina Shklovski, and Sameer Patil. "Empowering Resignation: There's an App for That". In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8096-6. url: https://doi.org/10.1145/3411764.3445293.

[188] Hamza Shaban. "An Amazon Echo Recorded a Family's Conversation, Then Sent It to a Random Person in Their Contacts, Report Says". In: *The Washington Post* (May 2018). url: https://www.washingtonpost.com/news/the-switch/wp/2018/05/24/an-amazon-echo-recorded-a-familys-conversation-then-sent-it-to-a-random-person-in-their-contacts-report-says/.

[189] Hamza Shaban. "An Amazon Echo Recorded a Family's Conversation, Then Sent It to a Random Person in Their Contacts, Report Says". In: *The Washington Post* (May 2018). url: https://www.washingtonpost.com/news/the-switch/wp/2018/05/24/an-amazon-echo-recorded-a-familys-

conversation-then-sent-it-to-a-random-person-in-their-contacts-report-says/.

[190] Bingyu Shen, Lili Wei, Chengcheng Xiang, Yudong Wu, Mingyao Shen, Yuanyuan Zhou, and Xinxin Jin. "Can Systems Explain Permissions Better? Understanding Users' Misperceptions under Smartphone Runtime Permission Model". In: *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 751–768. isbn: 978-1-939133-24-3. url: https://www.usenix.org/conference/usenixsecurity21/presentation/shen-bingyu.

[191] Faysal Hossain Shezan, Hang Hu, Jiamin Wang, Gang Wang, and Yuan Tian. "Read between the Lines: An Empirical Measurement of Sensitive Applications of Voice Personal Assistant Systems". In: *Proceedings of the Web Conference 2020*. WWW '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1006–1017. isbn: 978-1-4503-7023-3. doi: 10.1145/3366423.3380179. url: https://doi.org/10.1145/3366423.3380179.

[192] Yang Shi, Yang Wang, Ye Qi, John Chen, Xiaoyao Xu, and Kwan-Liu Ma. "IdeaWall: Improving Creative Collaboration through Combinatorial Visual Stimuli". In: *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 594–603. isbn: 978-1-4503-4335-0. doi: 10.1145/2998181.2998208. url: https://doi.org/10.1145/2998181.2998208.

[193] Tom Simonite. "Who's Listening When You Talk to Your Google Assistant?" In: *Wired* (Oct. 2019). url: https://www.wired.com/story/whos-listening-talk-google-assistant/.

[194] John M. Simpson. "Home Assistant Adopter Beware: Google, Amazon Digital Assistant Patents Reveal Plans for Mass Snooping". In: (2017). url: https://www.consumerwatchdog.org/privacy-technology/home-assistant-adopter-beware-google-amazon-digital-assistant-patents-reveal (visited on 04/04/2019).

[195] Manya Sleeper, Justin Cranshaw, Patrick Gage Kelley, Blase Ur, Alessandro Acquisti, Lorrie Faith Cranor, and Norman Sadeh. ""I Read My Twitter the next Morning and Was Astonished": A Conversational Perspective on Twitter Regrets". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2013, pp. 3277–3286. isbn: 978-1-4503-1899-0. url: https://doi.org/10.1145/2470654.2466448.

[196] Daniel J. Solove. "The Myth of the Privacy Paradox". en. In: *George Washington Law Review* 89 (Feb. 2020). issn: 1556-5068. doi: 10.2139/ssrn.3536265. url: https://www.ssrn.com/abstract=3536265 (visited on 11/04/2020).

[197] T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady. "explAIner: A Visual Analytics Framework for Interactive and Explainable Machine Learning". In: *IEEE Transactions on Visualization and Computer Graphics* 26.1 (Jan. 2020), pp. 1064–1074. issn: 1941-0506. doi: 10.1109/TVCG.2019.2934629.

[198] Statista. "Global Market Share Held by Leading Desktop Internet Browsers from January 2015 to December 2018". In: (). url: https://www.statista.com/statistics/544400/market-share-of-internet-browsers-desktop/.

[199] Art Swift. "A Matter of Life and Death: Why We Must Take IoT Flaws Seriously". In: *Infosecurity Magazine* (Oct. 2015). url: https://www.infosecurity-magazine.com/opinions/a-matter-of-life-and-death/.

[200] Madiha Tabassum, Tomasz Kosiński, Alisa Frik, Nathan Malkin, Primal Wijesekera, Serge Egelman, and Heather Richter Lipford. "Investigating Users' Preferences and Expectations for Always-Listening Voice Assistants". In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3.4 (Dec. 2019). doi: 10.1145/3369807. url: https://doi.org/10.1145/3369807.

[201] Chuanqi Tao, Hongjing Guo, and Zhiqiu Huang. "Identifying Security Issues for Mobile Applications Based on User Review Summarization". In: *Information and Software Technology* 122 (2020), p. 106290. issn: 0950-5849. doi: 10.1016/j.infsof.2020.106290. url: http://www.sciencedirect.com/science/article/pii/S0950584920300409.

[202] Welderufael B. Tesfay, Jetzabel Serna, and Kai Rannenberg. "PrivacyBot: Detecting Privacy Sensitive Information in Unstructured Texts". In: *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. Oct. 2019, pp. 53–60. doi: 10.1109/SNAMS.2019.8931855.

[203] Christian Tiefenau, Maximilian Häring, Eva Gerlitz, and Emanuel von Zezschwitz. "Making Privacy Graspable: Can We Nudge Users to Use Privacy Enhancing Techniques?" In: *arXiv:1911.07701 [cs]* (Nov. 2019). arXiv: 1911.07701 [cs]. url: http://arxiv.org/abs/1911.07701.

[204] Alexander Tsesis. "The Right to Erasure: Privacy, Data Brokers, and the Indefinite Retention of Data". In: *Wake Forest L. Rev.* 49 (2014), p. 433.

[205] Güliz Seray Tuncay, Jingyu Qian, and Carl A. Gunter. "See No Evil: Phishing for Permissions with False Transparency". In: *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 415–432. isbn: 978-1-939133-17-5. url: https://www.usenix.org/conference/usenixsecurity20/presentation/tuncay.

[206] Joseph Turian, Lev Ratinov, and Yoshua Bengio. "Word Representations: A Simple and General Method for Semi-Supervised Learning". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. ACL '10. USA: Association for Computational Linguistics, 2010, pp. 384–394.

[207] Joseph Turow. *The Voice Catchers: How Marketers Listen in to Exploit Your Feelings, Your Privacy, and Your Wallet*. New Haven: Yale University Press, 2021. isbn: 978-0-300-24803-6.

[208] Joseph Turow, Michael Hennessy, and Nora Draper. "Persistent Misperceptions: Americans' Misplaced Confidence in Privacy Policies, 2003–2015". In: *Journal of Broadcasting & Electronic Media* 62.3 (2018), pp. 461–478. doi: 10.1080/08838151.2018.1451867. eprint: https://doi.org/10.1080/08838151.2018.1451867. url: https://doi.org/10.1080/08838151.2018.1451867.

[209] U.S. Information Security Oversight Office. *Developing and Using Security Classification Guides*. Oct. 2018. url: https://www.archives.gov/files/isoo/training/scg-handbook.pdf.

[210] U.S. Office of the Director of National Intelligence. *Office of the Director of National Intelligence Classification Guide*. Sept. 2014. url: https://www.dni.gov/files/documents/FOIA/DF-2015-00044%20(Doc1).pdf.

[211] Tavish Vaidya, Yuankai Zhang, Micah Sherr, and Clay Shields. "Cocaine Noodles: Exploiting the Gap between Human and Machine Speech Recognition". In: *9th USENIX Workshop on Offensive Technologies (WOOT 15)*. Washington, D.C.: USENIX Association, Aug. 2015. url: https://www.usenix.org/conference/woot15/workshop-program/presentation/vaidya.

[212] Anthony Vance, Jeffrey L. Jenkins, Bonnie Brinton Anderson, Daniel K. Bjornn, and C. Brock Kirwan. "Tuning Out Security Warnings: A Longitudinal Examination of Habituation Through fMRI, Eye Tracking, and Field Experiments". In: *MIS Quarterly* 42.2 (Feb. 2018), pp. 355–380. issn: 02767783, 21629730. doi: 10.25300/MISQ/2018/14124. url: https://misq.org/skin/frontend/default/misq/pdf/appendices/2018/V42I2Appendices/01_14124_RA_VanceJenkins.pdf (visited on 08/04/2021).

[213] Anthony Vance, Brock Kirwan, Daniel Bjornn, Jeffrey Jenkins, and Bonnie Brinton Anderson. "What Do We Really Know about How Habituation to Warnings Occurs over Time? A Longitudinal FMRI Study of Habituation and Polymorphic Warnings". In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2017, pp. 2215–2227. isbn: 978-1-4503-4655-9. url: https://doi.org/10.1145/3025453.3025896.

[214] *Virginia Consumer Data Protection Act*. 2021. url: https://lis.virginia.gov/cgi-bin/legp604.exe?212+sum+HB2307.

[215] Sarah Theres Völkel, Daniel Buschek, Malin Eiband, Benjamin R. Cowan, and Heinrich Hussmann. "Eliciting and Analysing Users' Envisioned Dialogues with Perfect Voice Assistants". In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8096-6. doi: 10.1145/3411764.3445536. url: https://doi.org/10.1145/3411764.3445536.

[216] Sarah Theres Völkel, Renate Haeuslschmid, Anna Werner, Heinrich Hussmann, and Andreas Butz. "How to Trick AI: Users' Strategies for Protecting Themselves from Automatic Personality Assessment". In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1–15. isbn: 978-1-4503-6708-0. doi: 10.1145/3313831.3376877. url: https://doi.org/10.1145/3313831.3376877.

[217] T. Walshe and A. Simpson. "An Empirical Study of Bug Bounty Programs". In: *2020 IEEE 2nd International Workshop on Intelligent Bug Fixing (IBF)*. Feb. 2020, pp. 35–44. doi: 10.1109/IBF50092.2020.9034828.

[218] R. Wang, Z. Wang, B. Tang, L. Zhao, and L. Wang. "SmartPI: Understanding Permission Implications of Android Apps from User Reviews". In: *IEEE Transactions on Mobile Computing* (2019), pp. 1–1. issn: 1558-0660. doi: 10.1109/TMC.2019.2934441.

[219] David Watkins. *Global Smart Speaker Vendor & OS Shipment and Installed Base Market Share by Region: Q4 2018*. Tech. rep. StrategyAnalytics, Feb. 2019. url: https://www.strategyanalytics.com/access-services/devices/connected-home/smart-speakers-and-screens/market-data/report-detail/global-smart-speaker-vendor-os-shipment-and-installed-base-market-share-by-region-q4-2018.

[220] Jing Wei, Tilman Dingler, Enying Gong, Brian Oldenburg, and Vassilis Kostakos. "Proactive Smart Speakers for Chronic Disease Management: Challenges and Opportunities". In: *"Mapping Grand Challenges for the Conversational User Interface Community" workshop, CHI 2020* (2020). url: http://www.speechinteraction.org/CHI2020/papers/Proactive%20Smart%20Speakers%20for%20Chronic%20Disease%20Management(2).pdf.

[221] Jing Wei, Tilman Dingler, and Vassilis Kostakos. "Developing the Proactive Speaker Prototype Based on Google Home". In: *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI EA '21. New York, NY, USA: Association for Computing Machinery, 2021. isbn: 978-1-4503-8095-9. doi: 10.1145/3411763.3451642. url: https://doi.org/10.1145/3411763.3451642.

[222] Karen Weise. "Hey, Alexa, Why Is Amazon Making a Microwave?" In: *The New York Times* (2018). url: https://www.nytimes.com/2018/09/20/technology/amazon-alexa-new-features-products.html.

[223] Christian Werner. "Explainable AI through Rule-Based Interactive Conversation." In: *EDBT/ICDT Workshops*. 2020.

[224] Primal Wijesekera, Arjun Baokar, Ashkan Hosseini, Serge Egelman, David Wagner, and Konstantin Beznosov. "Android Permissions Remystified: A Field Study on Contextual Integrity". In: *24th USENIX Security Symposium (USENIX Security 15)*. Washington, D.C.: USENIX Association, Aug. 2015, pp. 499–514. isbn: 978-1-931971-23-2. url: https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/wijesekera.

[225] Primal Wijesekera, Arjun Baokar, Lynn Tsai, Joel Reardon, Serge Egelman, David Wagner, and Konstantin Beznosov. "The Feasibility of Dynamically Granted Permissions: Aligning Mobile Privacy with User Preferences". In: *2017 IEEE Symposium on Security and Privacy (SP)*. May 2017, pp. 1077–1093. doi: 10.1109/SP.2017.51.

[226] Molly Wood. "Google Buzz: Privacy Nightmare". In: *CNET News* (Feb. 2010). url: https://www.cnet.com/news/google-buzz-privacy-nightmare/.

[227] Rayoung Yang and Mark W. Newman. "Learning from a Learning Thermostat: Lessons for Intelligent Systems for the Home". In: *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. UbiComp '13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 93–102. isbn: 978-1-4503-1770-2. doi: 10.1145/2493432.2493489. url: https://doi.org/10.1145/2493432.2493489.

[228] Yaxing Yao, Justin Reed Basdeo, Oriana Rosata Mcdonough, and Yang Wang. "Privacy Perceptions and Designs of Bystanders in Smart Homes". In: *Proc. ACM Hum.-Comput. Interact.* 3.CSCW (Nov. 2019). doi: 10.1145/3359161. url: https://doi.org/10.1145/3359161.

[229] Bob Yirka. "Google Nest Hacker Finds Evidence of Google Considering Getting Rid of 'Hey Google' Hot Words". In: *Tech Xplore* (Oct. 2020). url: https://techxplore.com/news/2020-10-google-hacker-evidence-hey-hot.html.

[230] 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)*. Santa Clara, CA: USENIX Association, 2017, pp. 65–80. isbn: 978-1-931971-39-3. url: https://www.usenix.org/conference/soups2017/technical-sessions/presentation/zeng.

[231] Serena Zheng, Noah Apthorpe, Marshini Chetty, and Nick Feamster. "User Perceptions of Smart Home IoT Privacy". In: *Proc. ACM Hum.-Comput. Interact.* 2.CSCW (Nov. 2018), 200:1–200:20. issn: 2573-0142. doi: 10.1145/3274469. url: http://doi.acm.org/10.1145/3274469.

[232] Marrian Zhou. "Amazon's Alexa Guard Can Alert You If an Echo Detects Smoke Alarm, Breaking Glass". In: *CNET News* (Dec. 2018). url: https://www.cnet.com/news/amazons-alexa-guard-can-alert-you-if-an-echo-detects-smoke-alarm-breaking-glass/.

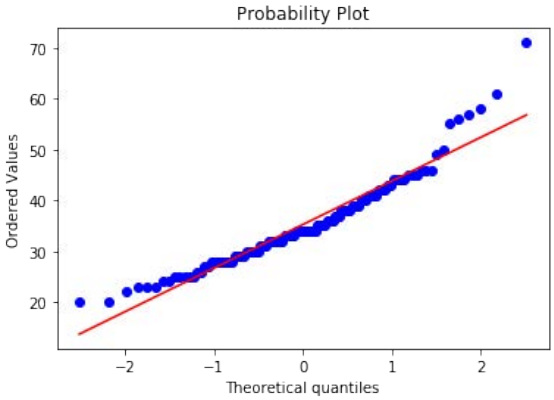
Appendix A

Supplemental Materials for Smart Speakers Study (Chapter 4)

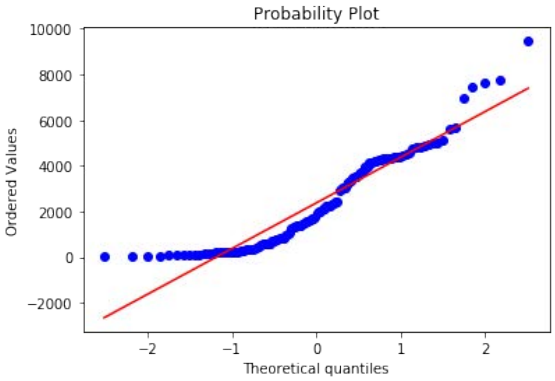
A.1 QQ plots

Before using the t-test to compare age and number of responses across different subgroups, we used a Q-Q plot to verify that the data was approximately normally distributed. The plots appear below.

Q-Q plot of participants' age



Q-Q plot of number of interactions obtained from each participant



A.2 Survey instrument

Please select which kind of smart speaker you have. (If you have both, please select the one you use the most.)

- I have a smart speaker with Amazon Alexa (such as Amazon Echo, Echo Dot, etc.)
- I have a smart speaker with Google's Assistant (such as Google Home, Home Mini, etc.).
- I don't have either kind of smart speaker

Approximately when did you start using this smart speaker?

How many people are in your household?

Who in your household has the Amazon Alexa / Google Home app installed on their mobile device and/or linked to the main Amazon/Google account?

- Only me
- Myself and some of the other members of the household
- Every member of the household
- Someone else in the household, but not me
- Not sure

The extension will now test your eligibility for our study. As a reminder, to be eligible for our study, you need to meet the following criteria:

- You've owned your <*Device*> for at least 1 month.

- You've used it at least 30 times.
- The <*Device*> is linked to your <*Company*> account (so that you're able to access and read its settings).

When you click "Continue," the extension will automatically access your account to verify these criteria. To do this, we'll open a page from <*Company*> in the background and get the information from there. If you're not logged in, we'll ask you to open a new tab and log in to <*Company*> as you would normally. At no point in our study will we have access to your password and nothing from your account (other than the eligibility information) will ever be shared with us.

After you ask <*Assistant*> a question or say a command, what do you believe happens to the audio of your interaction?

- It doesn't get saved at all
- It gets saved temporarily
- It gets saved indefinitely
- I don't know

Did you know that there is a page on <*Company*>'s website where you can see the recordings and transcripts of all your past interactions with your <*Device*>?

- Yes
- No

Which statement best describes your use of the review feature?

- I know I can review my interactions, but don't know how to do it.
- I know how to review my interactions, but have never done it.
- I've reviewed my interactions on individual occasions.
- I regularly review my interactions.
- I didn't know I could review my interactions.

What usually prompts you to review your interactions?

When you were previously reviewing your device's activity, did you encounter interactions that were initiated by someone other than yourself? (i.e., it wasn't you talking to <*Device*>, it was your partner, child, friend, etc.)

- Yes
- No

Have you ever discussed with another home occupant or visitor a recording of them that you listened to? What prompted you, and how did the conversation go?

Do you believe anyone else in your household has access to the recordings made by your <*Device*>?

- Yes
- No

How would you feel about other members of your household reviewing your interactions with <*Device*>?

Were you aware that the review interface allows you to delete specific recordings of you interacting with your <*Device*>?

- Yes
- No

Which statement best describes your use of the deletion feature?

- I knew you could delete recordings, but have never done this.
- I've deleted recordings on individual occasions.
- I regularly delete recordings.
- I didn't know you could delete recordings.
- What usually prompts you to delete interactions?

How did you decide not to delete any interactions?

We'll now ask you a few questions about up to 5 specific interactions you've had with <*Assistant*>. The browser extension you've installed will access a random recording from <*Company*>'s website and show it to you. It will not be sent to the research team.

Here is a recording and transcript of a question (or instruction) you asked <*Assistant*>. Please listen to it before answering the following questions.

(Reminder: we will never hear this recording or see its transcription. Only your answers to the questions below are transmitted to the research team.)

Who is (primarily) speaking in this recording?

- This is a recording of me.
- This is a recording of someone else in my household.
- This is a recording of a guest.
- This is a recording of the TV, music, or other pre-recorded audio.
- This is a recording of noise/gibberish.
- This is a legitimate recording or transcript, but I'm not sure who said it.
- Other

Please describe what was said to <*Assistant*> in this recording. i.e., what were you (or the person speaking) asking <*Assistant*>?
If you are comfortable, feel free to paste the transcript of your interaction. Otherwise, please provide a general description.
If you're not comfortable sharing any details of this interaction, please write down "I'd rather not say."
If <*Assistant*> misunderstood the question or command, describe the **intended** request.

Did you (or the person speaking) address <*Assistant*>, or was the audio recorded by accident?

- I was/they were speaking to <*Assistant*>.
- It was an accident.

Do you remember asking this question/making this request?

- Yes
- No
- Not sure

How would you feel if this audio recording were stored for the following periods of time (after the recording takes place)?
(*5-point Likert scale: Completely acceptable, Somewhat acceptable, Neutral, Somewhat unacceptable, Completely unacceptable*)

- Just long enough to complete your request
- For one week
- For one year
- For an unspecified period of time

- Forever
- As long as you own the device and use the service
- As long as any party other than the manufacturer cannot access it

How acceptable would it be for this audio recording to be processed and analyzed by...
(*5-point Likert scale, acceptability*)

- A computer program performing quality control for <*Company*>?
- A human, working for <*Company*>, performing quality control?

How would you feel if <*Company*> used this **audio recording** for...
(*5-point Likert scale, acceptability*)

- Improving <*Assistant*>'s performance, functions, or services
- Providing you with additional functionality powered by <*Company*>
- Providing you with promotional offers from <*Company*>
- Providing you with additional functionality powered by other companies
- Providing you with promotional offers from other companies

How would you feel if <*Company*> used only the the transcript (not the recording) of this interaction for...
(*5-point Likert scale, acceptability*)

- Improving <*Assistant*>'s performance, functions, or services
- Providing you with additional functionality powered by <*Company*>
- Providing you with promotional offers from <*Company*>
- Providing you with additional functionality powered by other companies
- Providing you with promotional offers from other companies

Given the option, would you delete this specific recording from <*Company*>'s servers?

- Yes
- No

Why or why not?

In your opinion, how long should <*Company*> store your data before deleting it?

- Until you manually delete it

- As long as the company wants to
- _____ hours
- _____ days
- _____ weeks
- _____ months
- _____ years

Suppose a third-party tool (such as a browser extension) were available, which would automatically delete your recordings from <*Company*> after a certain amount of time (that you specify). Would you install this tool?
(*5-point Likert scale: Very likely, Likely, Neither likely nor unlikely, Unlikely, Very unlikely*)

Why or why not?

Suppose <*Company*> had the option to automatically delete your recordings after a certain amount of time (that you specify). Do you believe you would enable this feature?
(*5-point Likert scale, likelihood*)

Why or why not?

Have you ever asked <*Assistant*> a question/command that you wish you could delete due to privacy concerns? What about it was sensitive?

Suppose <*Assistant*> had a feature that let you automatically screen out certain recordings and prevent them from being saved. Do you believe you would ever make use of this feature?
(*5-point Likert scale, likelihood*)

Suppose, as in the previous question, that <*Assistant*> had a feature that let you automatically screen out certain recordings and prevent them from being saved. Based on **your** privacy concerns, which characteristics would you want it to screen on (if any)? (Consider topics, people speaking, time of day, usage patterns, or other categories)

In the past, have you had any privacy concerns about your <*Device*>? Please tell us about them.

In the past, did you take any steps to protect your privacy when using your <*Device*>? What were they?

In the future, do you intend to take any steps to protect your privacy when using your <*Device*>? What do you plan to do?

If <*Company*> said that they were using your recordings to "improve <*Device*>'s performance, functions, or services," what do you think that would mean?

Is there anything else you'd like to share with the researchers about your experience with your <*Device*>? If so, please tell us below.

Do you have any feedback for us about how this study went? (optional)

What is your gender?

- Female
- Male
- Other
- Prefer not to say

What is your age?

Appendix B

Supplemental Materials for Architecture Groundwork (Chapter 5)

Here, we provide a range of examples of always-listening apps; each is illustrated by sample interactions that may trigger them or a general description of the speech they may find relevant. We organize the apps, loosely, based on the setting where they are most likely to be invoked.

Living room

Q&A / search

Queries you would ask Google/Siri/etc. today

- Does anyone know who was president in 1837?

Add to calendar

Automatically add planned meetings and appointments to the user's calendar.

- Let's plan to get lunch at noon next Thursday.
- I took the doctor's appointment at 10 AM tomorrow. I'll pick you up at 9 and we'll go together.

Remind later

Detect when the user wants to remember something and automatically save it, in order to remind them at an appropriate time.

- I need to remember to water the plants on Friday.
- Remind me to call Morgan!
- Don't forget that Casey has soccer practice on Wednesday.
- The school needs Blake's permission slip back by Thursday.

Timer

- Start a countdown for 60 seconds
- Can you time how long it takes me to do 10 pull-ups?

Remember locations, send to map

Take note of locations mentioned during a conversation, so that they can be highlighted on the user's personal map (e.g., in a smartphone app).

- I should get stamps when I'm in the store tomorrow.
- "This cake is amazing, where did you get it?" "Virginia Bakery" "I should remember to stop by next time I'm in Berkeley"

Learn book/TV/movie/music preferences and make recommendations

- "1984 is the best book I've ever read." "I don't know, I kind of liked Brave New World more."
- "Do you listen to rap much?" "Not a lot. But it's fun to listen when my friends come over during the weekend or at a party.

Email client

- "Read me the email from Jack that I got this morning"
- "Did I get any new messages?"

Shazam

- What's the song playing in the background of this commercial?
- That song was dope! I want to look up the artist later.

Unified music history

Keep a record of any music playing, so that users can recall that information at a later point, or so that it can be used for offering personalized music recommendations.

- Do you remember the name of the track you played for me last night?

Change music based on mood in the room

While active, the app could keep track of any (non-sensitive) conversations in the room, perform sentiment analysis on them, then select the next track based on the overall mood of the room.

Read books out loud

- Hey AI, read me a book.
- Stop. Hold on.
- Can you repeat the last paragraph?
- Let's start over from the beginning of the chapter.

Trivia quiz

- I want to play a game.
- Okay, here's a trivia question for you. What's the largest city in Yukon?
- I'm not sure, I need a hint.
- Okay! It's also the smallest city in Yukon.

Language practice

When a user wants to practice their foreign language skills, the app could serve as their conversation partner, or suggest vocabulary and correct grammar mistakes as the user is talking.

Swear jar

- That's the third time you've said <expletive> today. Your swear jar is up to $3 now.

Reservation maker

The app would initiate appointments and reservations on behalf of the user.

- "Let's celebrate your birthday at Fentons next week." "Alright, we need a reservation for 6 people"

Weather

The app could directly answer queries about the weather, as well as detect when a conversation discusses plans that may be affected by the weather, and offer up the relevant forecast.

- Will it rain tomorrow?
- Do you think I need a coat?
- Is August a good time to go to Thailand?
- [discussing a trip] Are you sure you want to go hiking this weekend? The forecast calls for rain.

Trip/travel recommendations

- I really want to get away to some place quiet with a beach for Memorial Day weekend. Can you think of anywhere? I don't want to pay more than a couple hundred dollars for tickets.

Health advice

- You seem to be coughing a lot today. The air quality isn't great; there's a lot of pollen. Would you like us to order you some allergy medicine?

Fitness monitor

- We've noticed you've been sitting for a while. Do you want to stand up and walk around for a couple of minutes?

Gift recommendations

- These headphones seem great, I want to get these someday.

Track baby's vocabulary

- Cat is your baby's third unique word!

Fact checker

The app could analyze, in real time, the veracity of statements made on TV (or by the speakers on the room).

- [responding to statement on TV] Politifact has rated this claim as False.

Artificial memory: kids' names edition

The app would keep track of potentially useful information about friends and acquaintances, such as the names or birthdays of their children.

- "Do you have kids?" "Yeah, three daughters." "Oh wow, I didn't know that! How old are they?" "Well, Drizella is 30, Anastasia is 25, and Ella is 20."

Ikea helper

While you're trying to put together Ikea furniture, the app could listen, read instructions out loud, and offers help and suggestions.

Score keeper

The app could gamify household chores by keeping track of children's scores.

- 10 points for Gryffindor!

Kid monitor

Detect if a child is shouting, fighting with their sibling, or is "engaging in mischief" [135].

Motion detector

The app can detect people's presence in certain rooms of the house, then turn on the heat/AC/light there, while turning it off in other rooms.

Kitchen

Shopping list

- We should get eggs.
- Don't forget to get milk on your way home.
- We're almost out of butter.

Fridge monitor

Keep track of items in the refrigerator and when they are likely to expire.

- Oh yay, you got cauliflower. [Reminder that it goes bad after X days.]
- Do we need milk? No, this one is good for another week.
- What's for dinner today? Turkey and mashed potatoes. [remember that these ingredients were (probably) used up]

Recipe search

- How long do I boil eggs?
- How do I make ratatouille?
- Do we have everything we need for cookies?

Cooking advisor

- What temperature should I set this oven to for asparagus?
- How hot should the oil be for deep-frying?
- How long has this lasagna been in the oven?
- What is the temperature of a medium rare steak?
- How much sugar should I put in the cake mixer? I do not want it to be too sweet.

Unit conversion

- How many cups of flour is 200 grams?
- How many teaspoons is one tablespoon?
- What's 600 grams in ounces?

Is this thing still good?

- Oh no, I left out the ham this morning. Can I still eat it?

Coffee maker

The app could respond directly to instructions to make coffee, but also be more proactive, for example offering to brew coffee if a user simply mentions that they are tired.

Bedroom

Baby monitor

Inform parents if it hears their baby crying in other room

Alarm

- Wake me up tomorrow at 8.
- Time to get up!
- Do you want to snooze for another 10 minutes?

Coffee scheduler

- Have a cup ready to go when I wake up tomorrow.

Clothes recommendations

- What should I wear today? Maybe shorts? "It's 40 degrees outside; are you sure you want to wear shorts?"

Laundry reminder

- Hey, it's been a week since you've done laundry. You're almost out of socks.

Business/meetings

Action items

Summarizes action items from meeting

Equal time monitor

Makes sure everyone in the meeting has a chance to talk and doesn't monopolize the meeting

Language watchdog

Lets people know when they could have used more inclusive language

Measuring interruptibility

Determines whether now is a good time for a phone call or other interruption, based on content/tone of the conversation

Appendix C

Supplemental Materials for Runtime Permissions Study (Chapter 7)

C.1 Conversation prompts

For each of the three rounds of the study (§7.3), participants were given a different prompt to guide their conversation with their partner. This section includes the specific directions provided to the participants, as well as the list of apps that was "active" for that conversation. In verbal instructions, we explained that these were suggestions, rather than a script to follow, and that participants were free to deviate from them, as long as they stayed with the main topic.

Task 1

Dinner + shopping Your task is to arrange to cook dinner with your partner. You can decide things like:

- which day you'll be cooking
- who will be doing the cooking
- what you will cook
- what recipe you will use (feel free to find one online!)
- whether you have the necessary ingredients for the recipe
- which ingredients you need to buy
- where you'll go to buy those ingredients
- when you'll do that shopping

As you work on this task, Alva's apps may try to offer helpful suggestions on its screen or out loud.

Installed apps Here are some of the apps installed on your device:

- Supermarket helper
- Recipe search
- Shopping list
- Reminders
- Maps
- Calendar
- Social network

Task 2

Booking a weekend trip Your task is to plan an outing for this weekend with your partner. As part of your conversation, you might:

- Discuss availability and other conflicting events
- Discuss budget
- Choose destination
- Look up things to do
- Choose activities
- Look up directions
- Decide on where to eat
- Talk about whom you want to invite along

Installed apps

- Maps
- Calendar
- Social network
- Travel info
- Weather
- Flights (and other tickets)
- Lodging
- Coupons

Task 3

Booking a vacation Your task is to plan a vacation together with your partner. As part of your conversation, you might:

- Choose travel dates
- Discuss budget
- Choose destination
- Look up things to do
- Choose activities
- Search for tickets
- Pay for tickets (*do we want to simulate this? if so, how?*)
- Decide on where to stay

Installed apps

- Maps
- Calendar
- Social network
- Travel info
- Weather
- Flights (and other tickets)
- Lodging
- Coupons

C.2 Interview guide

Round 1 (ask-every-time)

General impressions

- Please give us your general impressions of being an Alva device user. What did you like about it? What did you dislike?

Why do people deny requests?

- I noticed you denied (or didn't approve) app _'s permission request. Can you explain why?

General feedback about permission prompts

- What did you think of Alva's permission requests (in general)?
  - Understandability
    - Were they clear or were they confusing?
    - Did they provide enough information?
  - Modality

    - Would you prefer to receive these requests in some other way?
    - What did you think about receiving them on the device's screen? (instead of on your phone, etc.)
  - Attention
    - Were the notifications effective at getting your attention?
    - Do you think, in a real situation, you'd notice or interact with these requests?
    - Would you want them to draw more attention to the notification? (e.g., louder noise) Or less?
  - Distractingness
    - Were the requests too distracting?
    - Do you think they should be more noticeable or less?

Round 2

Condition-specific UX questions

Learning

- Do you think Alva accurately learned your preferences? (Please explain.)
- Would you want your preferences learned in this way (if the learning were more accurate)?
- Did you (want to) review the decisions made by the learning?

***General* privacy questions about this *specific* condition**

For this next question, I want to remind you about how Alva works:

- if you don't install any apps, no one will hear anything.
- if you install any apps, any app you install on Alva has to go through these permission requests you just experienced.
- if deny an app's request, then no one learns about what you were saying.

With that in mind:

- Assuming you had an Alva, how willing would you be to install apps — either new ones or the ones from today — on it?
- Overall, how do you feel about your privacy with respect to Alva?
  - Do you feel that your privacy is adequately protected?
  - If not, why not? What scenario are you envisioning? What's missing?

Interview 2 / exit interview

Condition-specific UX questions

Rules/heuristics

- Did you (want to) review the decisions made by the rule?
- Did you regret your decision to make it a rule? Are there choices the rule made that you would've preferred it didn't?
- Would you have wanted a more (or less) restrictive rule? "only allow locations when I said _"
- (if no rule ever used) Why didn't you make use of the "always allow/deny" option?

Comparing Alva 1 vs 2

- How did the experiences of Alva 1 and Alva 2 compare for you?
  - Which Alva version does *each of* you prefer? Why?
  - Did you find the differences between the two Alva versions meaningful? (Please explain.) How strong is this preference? Is it only because I'm asking? Would you only use one of them, or you prefer one but it's not that big a deal?
  - What are the pros and cons of each version?
  - Did you prefer the user experience one or the other?
    - By user experience, I mean things like the frequency of notifications, how distracting it was, and other aspects we just discussed.
  - Do you trust one or the other more?

Asking about realism of #wizard-of-oz

- I want to ask *each of* you about whether your feelings and decisions would be different if you encountered this device in other situations.
  - Do you think your privacy concerns would be different with other apps or conversations?
    - Would you be comfortable having a conversation that involves sensitive topics, if you knew the apps from today's session would be listening (but they'd still have to request permission before sharing any data)?
    - Would you be comfortable using app that *required* access to more sensitive topics (e.g., advising on health or finances), but they'd still have to request permission before sharing, using the mechanisms you experience today?
  - Do you think you would feel differently if this were a real device instead of a prototype?

* You allowed some permission requests today. Would you have allowed them if this were a real device?
* If you were visiting a friend's house, and they had a working version of this device, how would you feel?

Appendix D

Supplemental Materials for Transparency Study (Chapter 8)

D.1 App descriptions

Table D.1 below lists the apps used in our study, with the complete descriptions and examples shown to participants.

D.2 Combined results

Data in the results section is separated in the interest of legibility and understanding. Here, the same data is presented side-by-side to facilitate easier comparisons.

Additional figures

Figure D.1 shows the cut-off on the x-axis, with the number of participants who perceived the attack app as malicious on the y-axis. The figure shows that, by the fifth participant, all groups had at least one person who detected the attack.

Perceptions vs discovery

We observe in 8.5 that the *discovery* metric is approximately similar to the *perception* metric. This is evident in Table D.3. However, things appear different in Table D.2: there is a large gap between the two. To understand this gap, we examined the specific instances that contributed to this metric: cases where a participant submitted an utterance that was an example of an attack, but they then did not perceive the app as malicious. Based on this analysis, we hypothesize that most of these occurred when an utterance had information of interest (e.g., a birthday or a doctor's appointment) but was still arguably relevant to

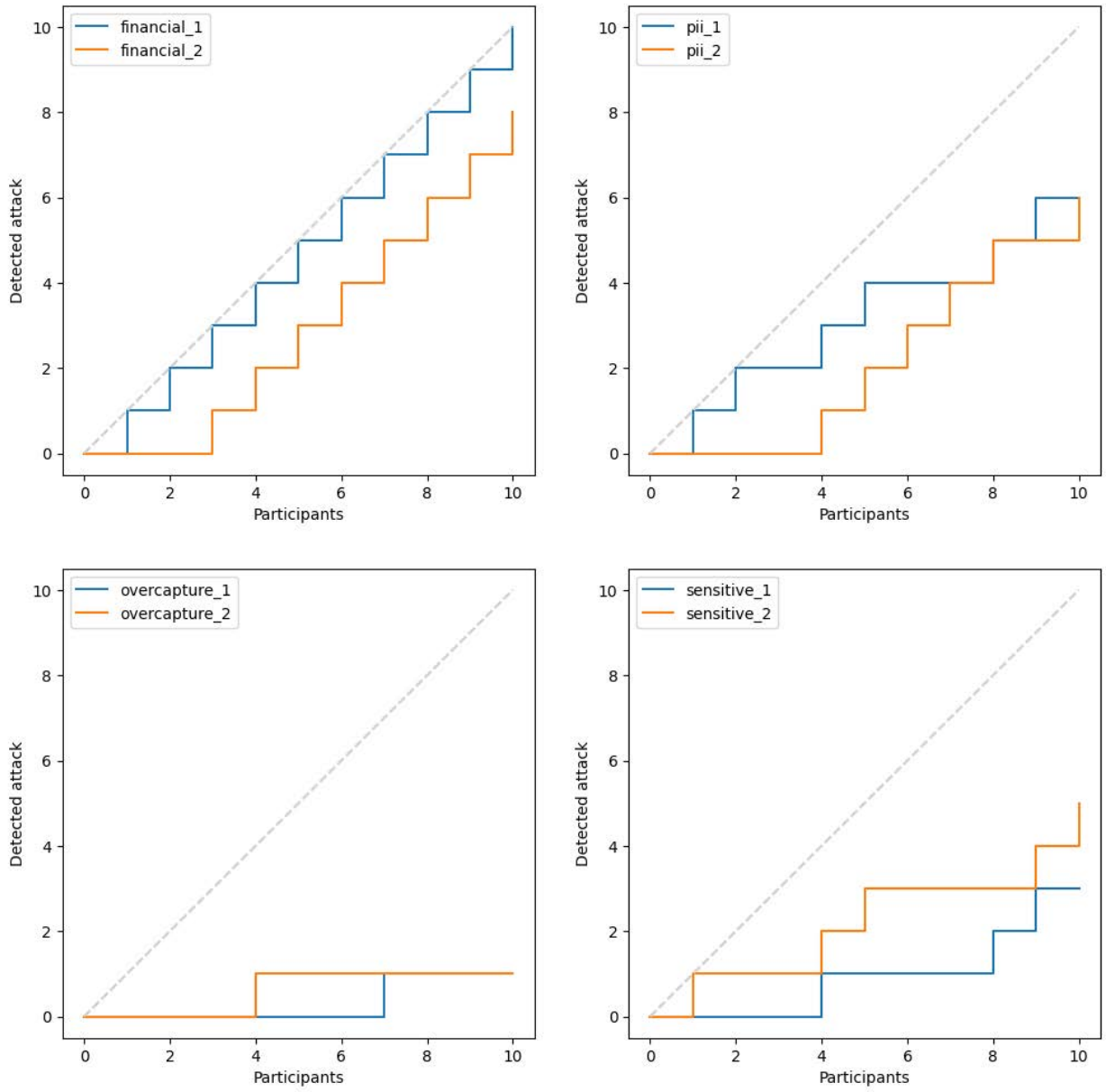
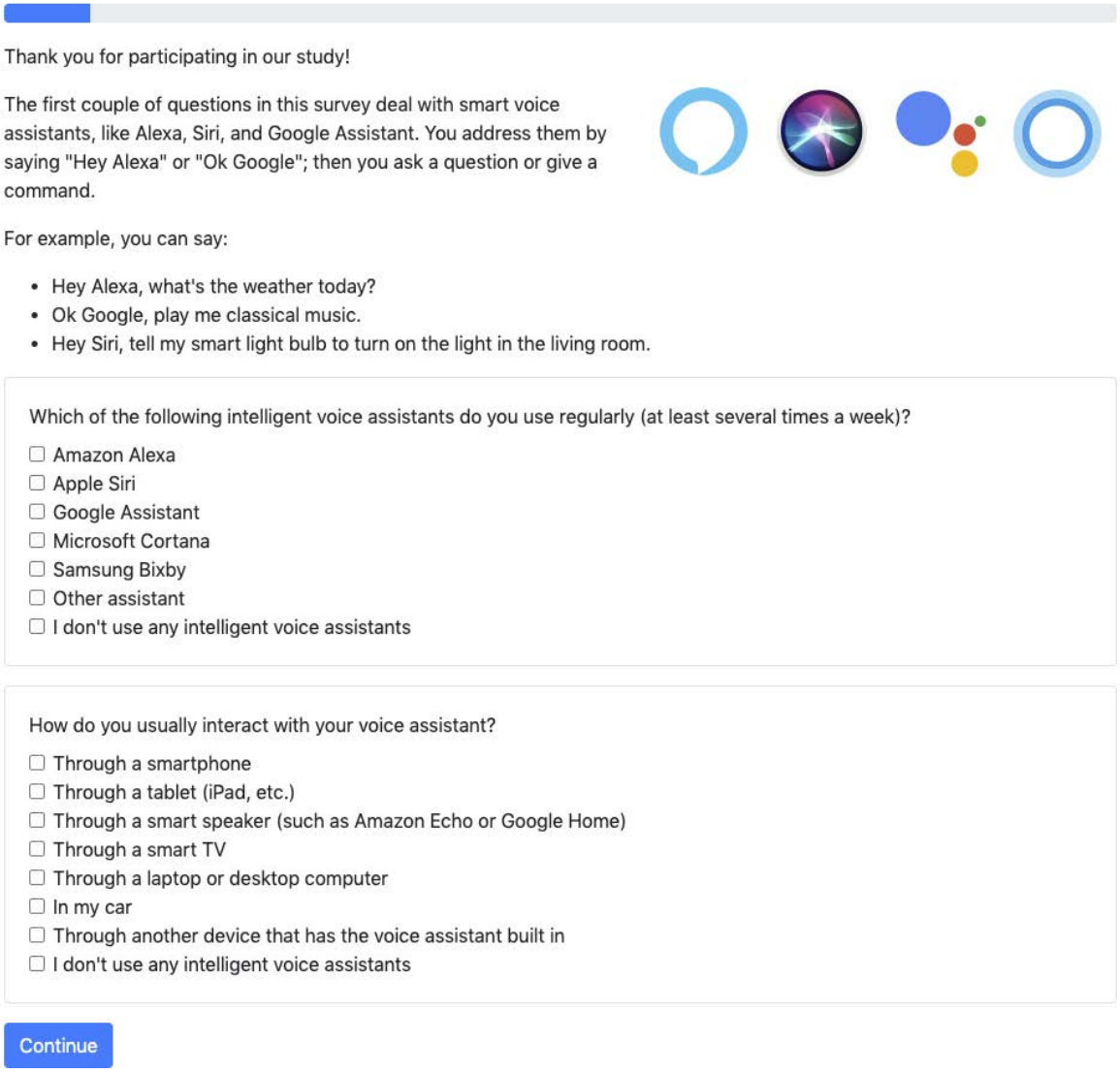


Figure D.1: **How many people would have detected the attack had the group size been smaller?**

the nature of the app (Reminders). This explains why the gap is much lower in apps seen post-treatment: for Weather and Cooking, there is much less room for ambiguity.

D.3 Survey instrument

As discussed in *Methods* (§8.4), our study had several variants, which differed slightly in the wording of the task formulation. The screenshots below detail one specific variant: collaborative mode.



The next couple of questions deal with smart speakers, like Amazon Echo and Google Home. If you're not familiar with them, they are wireless speakers integrated with intelligent voice assistants.

If you received a smart speaker as a gift, how likely would you be to set it up in your home and use it?

If you already have a smart speaker, assume that it stopped working, and you received the new one as a free replacement.

- ○ Very unlikely
- ○ Somewhat unlikely
- ○ Neutral
- ○ Somewhat likely
- ○ Very likely

If a close friend or family member received a smart speaker as a gift, how likely would you be to recommend that they set it up in their home and use it?

Assume that this friend or family member doesn't already have a smart speaker.

- ○ Very unlikely
- ○ Somewhat unlikely
- ○ Neutral
- ○ Somewhat likely
- ○ Very likely

Continue

Introducing Alva

We now ask you to **imagine that there's a new voice assistant** on the market: Alva. Unlike today's devices, you don't need to say specific words to wake up Alva, because it is always ready to help you. Alva can also provide services and suggestions based on conversations you have with other members of your household.

For example:

- Instead of saying "hey Siri, tell my smart light bulb, turn on," you could teach Alva to turn your lights on whenever you say "let there be light!"
- Alva could send you a message if it hears your smoke alarm going off.
- If you mention that you're going skiing this weekend, Alva could automatically add the weather at your destination to your phone's home screen.

Based on the description above, which of the following is true of the device in this survey?

- ○ The device reacts to your conversation only when explicitly addressed, for example when you say a specific word, such as "Alexa", "Siri", "Ok Google."
- ○ The device adds a video streaming feature and allows you to watch your favorite movies and shows on a big screen.
- ○ The device is waterproof and can be used in a bathroom or swimming pool.
- ○ The device is always on and can provide services and recommendations based on your current conversation, even if you don't say its name.

How useful do you think you would find Alva's functionality?

- ○ Not at all useful
- ○ Not very useful
- ○ Neutral
- ○ Somewhat useful
- ○ Very useful

Please explain your answer

Continue

Alva's apps

On your smartphone, most features are found in apps. If you want some new functionality, you can search the App Store for an app that provides it.

Alva works very similarly. Most features are also found in apps. If you want some new functionality, you can search the App Store for an app that provides it and download the new app.

Apps in the Alva App Store, like smartphone apps in Apple's App Store or Google Play, are usually created by third-party developers who are not affiliated with the manufacturers of Alva. Because of this, Alva cannot guarantee that apps work as described.

Attention check:

Based on the explanation above, which of the following is true of Alva?

- ○ A special vetting process guarantees that the behavior of each app matches its description.
- ○ All of the smart features are already built in to Alva. You never need to download anything new.
- ○ To add features to Alva, you need to download apps from the App Store. Apps are created by third-party developers.
- ○ I haven't read the explanation above about Alva and third-party apps.

Alva's Relevance Detection technology for apps

Since apps are created by third parties (and not verified by Alva), you might not want them to needlessly listen to all audio in your house. Instead, they should only hear the audio that's relevant to their functionality.

To minimize the audio shared with third-party apps, Alva relies on a special feature called Relevance Detection, which ensures that apps only get access to the audio that the app developers consider relevant. Here's how it works:

- Before an app gets to hear anything, all audio is pre-screened by the Alva device.
- To pre-screen, Alva applies an algorithm — the Relevance Detector — to the audio.
- If the Relevance Detector determines that some audio is relevant to the app, that audio will be shared with the app.
- If the Relevance Detector determines that some audio is *not* relevant to the app, that app will not be allowed to hear the irrelevant audio.
- An app's Relevance Detector is created by the developers of that app. The Alva App Store does not verify its correctness.
- This means some app's Relevance Detectors might not function as expected, resulting in them getting access to audio that's not relevant to their functionality.

Attention check:

Based on the explanation above, which of the following is true of Alva?

- ○ Relevance Detectors always function as expected, so apps never get access to audio that's not relevant to their functionality.
- ○ The manufacturers of Alva are the ones creating Relevance Detectors for each app.
- ○ App developers create their own Relevance Detectors and Alva does not verify their correctness.

Continue

Relevant or irrelevant?

Consider the following app, which has been submitted to the Alva App Store by a third party developer (i.e., someone who is **not affiliated with Alva**).

App name: Automatic Reminders

App description:
The purpose of this app is to automatically add to your calendar any appointments or reminders you mention out loud.

Example:
If you say "okay, it's settled, we'll meet next Thursday at noon," the app will add this meeting to your calendar.

Recall that Alva uses Relevance Detection to ensure that apps only get access to the audio that the app developers consider relevant. However, Alva does not independently verify the behavior of the Relevance Detector.

In the next couple of questions, we ask you to **read a sample conversation** between two people who are standing next to Alva and **consider whether the conversation is relevant to the app above**.

Person 1: Did you pack your lunch yet?
Person 2: No, I'm waiting until 11 o'clock, because I don't want it to sit out on the counter for too long.
Person 1: Okay, but don't forget! We need to leave at 11:30 sharp.

Based on the app description you just read, is this conversation relevant to the app's stated purpose?

- ○ Yes, the app could act on this conversation
- ○ No, this conversation is irrelevant to the app

Person 1: Hey, did you hear that *Parasite* won the Oscars this year?
Person 2: I didn't, but that's great, I loved that movie!
Person 1: Same here, I was so excited when it won!

Based on the app description you just read, is this conversation relevant to the app's stated purpose?

- ○ Yes, the app could act on this conversation
- ○ No, this conversation is irrelevant to the app

Continue

Note: the page below was not shown to participants in the *Install* variants.

Warning: malicious apps

As you learned previously, Alva does not verify the apps in the App Store. Unfortunately, this means that **some apps listen at inappropriate times** — not when you'd expect them to based on their description. This could happen by accident or the developer could do this on purpose, for example **to steal your personal information, to learn more about you for advertising, or with other intentions**.

What kind of information (or conversation topics) would you be worried about a malicious app hearing?

The good news is that **we have a way to identify malicious apps before they listen to any real person's audio**. To do this, we take an app for a *test drive* using its Relevance Detector.

Here's how it works:

- You enter some text. (It can be any length, representing an entire conversation.)
- The Relevance Detector tells us if the app would've heard this or not.
- Malicious apps can be identified because they hear things when they shouldn't be listening.

On the next page, **your task** will be to use this technique to find out whether an app is malicious or not.

Attention check:

Based on the explanation above, which of the following is true?

- ○ My task is to determine whether an app is malicious.
- ○ My task is to provide examples of things the app should hear.
- ○ My task is to finish this survey without paying attention to instructions.
- ○ My task is to decide whether or not I would install an app.

Continue

Is this app malicious?

As a reminder, Alva apps only get to hear audio their Relevance Detector deems relevant. The Alva App Store allows you to *test drive* an app by supplying any speech and seeing if the app would hear it.

Please test drive the Automatic Reminders app to **determine whether or not it is malicious**. Malicious apps are those that listen at inappropriate times in order to obtain personal information.

App name: Automatic Reminders

App description:
The purpose of this app is to automatically add to your calendar any appointments or reminders you mention out loud.

Example:
If you say "okay, it's settled, we'll meet next Thursday at noon," the app will add this meeting to your calendar.

Try enough inputs to make up your mind about whether this app is malicious or not. (It doesn't matter how well the app works, or whether you yourself would want to use it.)

Enter something you might say near Alva here:

Type a phrase, sentence, or entire conversation

Check now

We'll let you know if the app would hear this or not.

You said	Would the app hear it?
Speech you test drive will appear here	

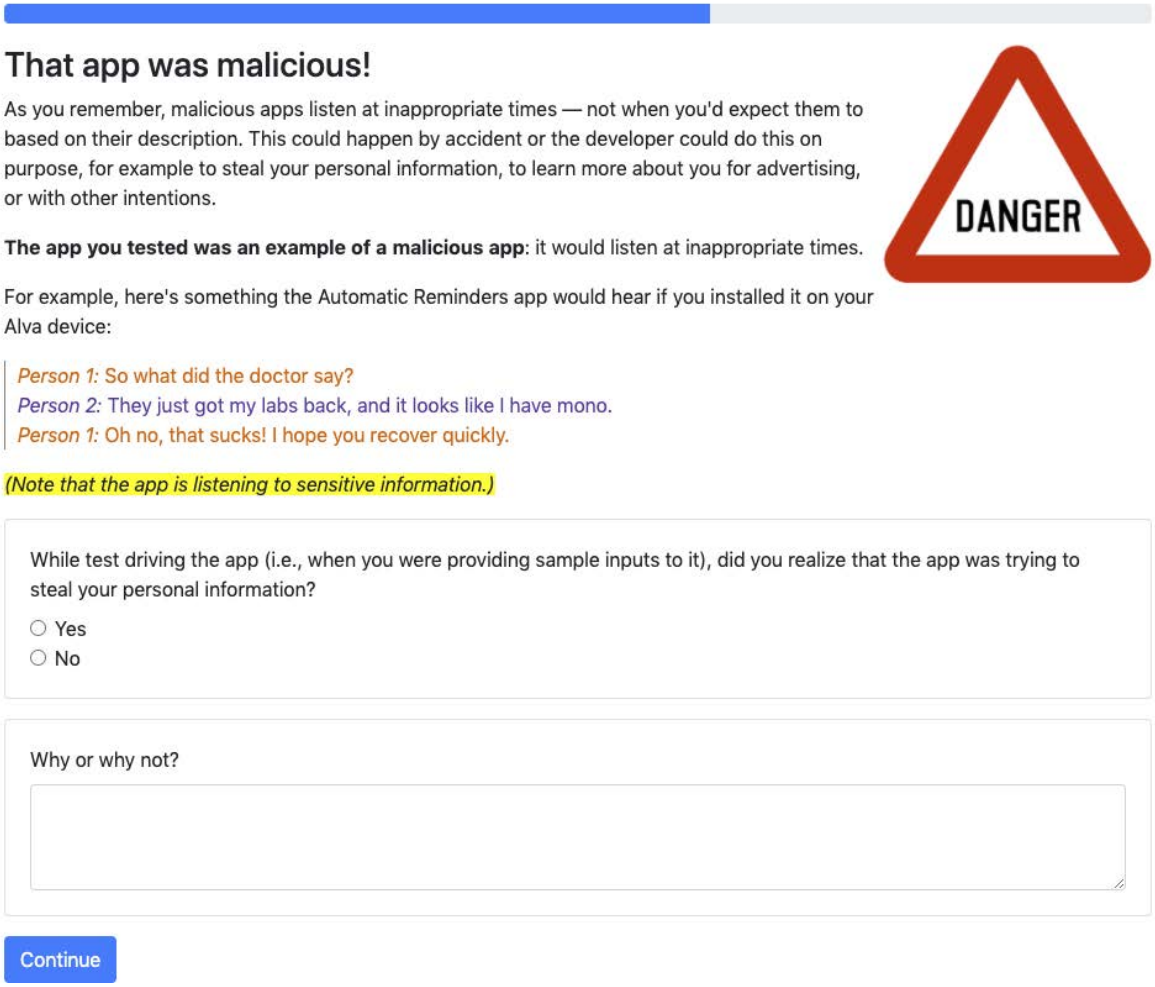
After you finish test driving this app, please tell us your conclusion.

Based on your experience test driving this app, how likely is it to be hearing things it shouldn't?

- ○ This app is *very likely* malicious (listening when it shouldn't be).
- ○ This app is *probably* malicious (listening when it shouldn't be).
- ○ I'm completely uncertain about whether or not this app is malicious or well-behaved.
- ○ This app is *probably* well-behaved (listening only when it's supposed to be).
- ○ This app is *very likely* well-behaved (listening only when it's supposed to be).

Please explain how you decided

Please try a few inputs before continuing



Note: participants in *Individual mode* did not see the *Report malicious behavior* button.

Test drive another app

Now that you know more about test driving apps, please test drive a new app. **Your goal**, once again, is to determine whether this app is malicious.

App name: Ambient Weather

App description:
The purpose of this app is to keep your phone's weather app updated with any destinations you mention in conversation.

Example:
If you're discussing your upcoming ski trip, the app will ensure that your phone's weather widget will show that location. You can also ask it questions directly ("what's the weather in Tahoe?").

Your Test Drive + Others' Test Drives

In this round, you can see the results of other users' Test Drives in addition to your own.
To help surface malicious apps, you can report bad behavior by clicking on the button next to each example.

Someone said:	**Would the app hear it?**	
xn7n1s1r41	The app would hear this.	Report malicious behavior 1
14r452p3xvbh	The app would hear this.	Report malicious behavior 1
8flae38dgaf	The app would hear this.	Report malicious behavior
n4qwjdxrjq	The app would hear this.	Report malicious behavior
gpb6xoenn6o	The app would hear this.	Report malicious behavior

Enter something you might say near Alva here:

Type a phrase, sentence, or entire conversation

Check now

We'll let you know if the app would hear this or not.

After you finish test driving this app, please tell us your conclusion.

Based on your experience test driving this app, how likely is it to be hearing things it shouldn't?

- ○ This app is *very likely* malicious (listening when it shouldn't be).
- ○ This app is *probably* malicious (listening when it shouldn't be).
- ○ I'm completely uncertain about whether or not this app is malicious or well-behaved.
- ○ This app is *probably* well-behaved (listening only when it's supposed to be).
- ○ This app is *very likely* well-behaved (listening only when it's supposed to be).

Please explain how you decided

Test drive one more app

One more time, please! Please test drive the following app to **determine whether it is malicious**.

App name: Chef of the Future

App description:
The purpose of this app is to advise you on any questions that come up in the kitchen.

Example:
You can ask "Chef" about what goes into recipes, which ingredients you can substitute for others, or for other advice about cooking. If it hears your question ("oh no, I think I added a tablespoon of salt instead of a teaspoon!") it'll remember what you were cooking and advise you accordingly ("don't worry! just add one more cup of water").

Your Test Drive + Others' Test Drives

Once again, you can see the results of other users' Test Drives in addition to your own.
To help surface malicious apps, you can report bad behavior by clicking on the button next to each example.

Someone said:	**Would the app hear it?**	
You're the first person to test drive this app. Other users will see the results of your test drive.		

Enter something you might say near Alva here:

Type a phrase, sentence, or entire conversation

Check now

We'll let you know if the app would hear this or not.

After you finish test driving this app, please tell us your conclusion.

Based on your experience test driving this app, how likely is it to be hearing things it shouldn't?

- ○ This app is *very likely* malicious (listening when it shouldn't be).
- ○ This app is *probably* malicious (listening when it shouldn't be).
- ○ I'm completely uncertain about whether or not this app is malicious or well-behaved.
- ○ This app is *probably* well-behaved (listening only when it's supposed to be).
- ○ This app is *very likely* well-behaved (listening only when it's supposed to be).

Please explain how you decided

Please try a few inputs before continuing

Test Drive reflections

Thank you for learning about Alva and taking a few apps for a Test Drive. We now have a couple more questions for you, about your experience overall.

What strategy did you use to test whether an app is malicious?

Was your testing strategy different when you were able to see other people's Test Drives? Please explain.

How useful did you find the Test Drive interface for the purpose of identifying malicious apps?

- ○ Very useless
- ○ Somewhat useless
- ○ Neutral
- ○ Somewhat useful
- ○ Very useful

Suppose you owned Alva and were considering installing an app. Would you use the Test Drive interface to help you decide? Please explain how you'd use it or why you wouldn't.

Continue

How would you test if this app is malicious?

We have a final task for you, but it's a bit different from the Test Drives you just did.

For the app below, we don't have a Relevance Detector, so you can't Test Drive it. However, we're asking you to submit inputs you would use in a Test Drive to **find out if the app is malicious**.

App name: What should I watch next?

App description:
This app keeps track of the movies/TV shows/videos you watch, and the opinions you expressed about them. Then when you ask it, "what should I watch next?", it can provide a recommendation for you.

Example:
"Hey, did you hear that Parasite won the Oscars this year?" "I didn't, but that's great, I loved that movie!" "Same here, I was so excited when it won!" (If the app heard this conversation, it would recommend films similar to the one mentioned.)

As a reminder, malicious apps listen at inappropriate times — not when you'd expect them to based on their description — and try to find out information about you that you might not want them to know.

For example, here's something a malicious app would hear if you installed it on your Alva device:

> *Person 1:* So what did the doctor say?
> *Person 2:* They just got my labs back, and it looks like I have mono.
> *Person 1:* Oh no, that sucks! I hope you recover quickly.

(Note that the app is listening to sensitive information.)

Please enter at least 5 inputs you would use to find out if the app above is malicious.
(It doesn't matter how well the app works, or whether you yourself would want to use it.)

Type a phrase, sentence, or entire conversation

Submit

Enter something that, if the app heard this audio, you would consider it malicious.

Speech heard
Speech you submit will appear here

Please enter at least 5 inputs to continue

Almost finished! To help us understand our participants, please tell us a couple of things about yourself.

What is your gender?

What is your age?

How many people are in your household?

How many children under the age of 18 are in your household?

Please share any comments about Alva or any feedback you have for us about this study.

Continue

Table D.1: **Apps** shown during the Test Drives to participants.

	Name	Description	Example
reminders	Automatic Reminders	The purpose of this app is to automatically add to your calendar any appointments or reminders you mention out loud.	If you say "okay, it's settled, we'll meet next Thursday at noon," the app will add this meeting to your calendar.
cooking	Chef of the Future	The purpose of this app is to advise you on any questions that come up in the kitchen.	You can ask "Chef" about what goes into recipes, which ingredients you can substitute for others, or for other advice about cooking. If it hears your question ("oh no, I think I added a tablespoon of salt instead of a teaspoon!") it'll remember what you were cooking and advise you accordingly ("don't worry! just add one more cup of water").
weather	Ambient Weather	The purpose of this app is to keep your phone's weather app updated with any destinations you mention in conversation.	If you're discussing your upcoming ski trip, the app will ensure that your phone's weather widget will show that location. You can also ask it questions directly ("what's the weather in Tahoe?").
movie rec.s	What should I watch next?	This app keeps track of the movies/TV shows/videos you watch, and the opinions you expressed about them. Then when you ask it, "what should I watch next?", it can provide a recommendation for you.	"Hey, did you hear that Parasite won the Oscars this year?" "I didn't, but that's great, I loved that movie!" "Same here, I was so excited when it won!" (If the app heard this conversation, it would recommend films similar to the one mentioned.)

Table D.2: **Pre-treatment**: percentage of participants who *perceived* the app as malicious and *discovered* malicious utterances (further clarified in 8.5), separated between the *Install* and *Test* variants, and further broken down by attack condition.

	Perceived	Discovered
Install ($n = 60$)	*N/A*	15%
Financial ($n = 22$)	*N/A*	9.1%
Sensitive ($n = 11$)	*N/A*	36%
PII ($n = 18$)	*N/A*	17%
Overcapture ($n = 9$)	*N/A*	0%
Test ($n = 140$)	26%	46%
Financial ($n = 38$)	45%	47%
Sensitive ($n = 34$)	32%	65%
PII ($n = 38$)	16%	42%
Overcapture ($n = 30$)	10%	27%

Table D.3: **Post-treatment**: percentage of participants who *perceived* the attack app as malicious and *discovered* malicious utterances (further clarified in 8.5), separated between the *Install* and *Test* variants, and further broken down by attack condition.

	Perceived	Discovered
Install ($n = 60$)	20%	15%
Financial ($n = 14$)	21%	14%
Sensitive ($n = 13$)	31%	31%
PII ($n = 18$)	17%	17%
Overcapture ($n = 15$)	13%	0%
Test ($n = 60$)	45%	50%
Financial ($n = 16$)	75%	75%
Sensitive ($n = 18$)	44%	50%
PII ($n = 13$)	46%	54%
Overcapture ($n = 13$)	8%	15%

Table D.4: **False positives**: fraction of participants who, post-treatment, *perceived* the benign app as malicious, separated between the *Install* and *Test* variants.

	Perceived
Install ($n = 60$)	10%
Test ($n = 60$)	6.7%

You have reached the end of this dissertation. Congratulations on your tenacity!