AN EXPERIMENTAL EVALUATION OF SMART TOYS' SECURITY AND PRIVACY PRACTICES

MOUSTAFA MAHMOUD

A THESIS

IN

THE CONCORDIA INSTITUTE FOR INFORMATION SYSTEMS ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF APPLIED SCIENCE IN INFORMATION SYSTEMS

SECURITY

CONCORDIA UNIVERSITY

MONTRÉAL, QUÉBEC, CANADA

MARCH 2018

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Moustafa Mahmoud**

Entitled: **An Experimental Evaluation of Smart Toys' Security and Privacy Practices**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science in Information Systems Security

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. W. Lucia ________________________ Chair

Dr. A. Youssef ________________________ Supervisor

Dr. M. Mannan ________________________ CIISE Examiner

Dr. W. Hamouda ________________________ External Examiner (ECE)

Approved ________________________________

Chair of Department or Graduate Program Director

__________ 20 ______ ________________________

Dr. Amir Asif, Dean

Faculty of Engineering and Computer Science

ABSTRACT

An Experimental Evaluation of Smart Toys' Security and Privacy Practices

Moustafa Mahmoud

Smart toys have captured an increasing share of the toy market, and are growing ubiquitous in households with children. These toys can be considered as a subset of Internet of Things (IoT) devices, often containing sensors and artificial intelligence capabilities. They may collect personal information, and frequently have Internet connectivity directly or indirectly through companion apps. Recent studies have found security flaws in many smart toys that have led to serious privacy leaks or allowed tracking a child's physical location. Some well-publicized discoveries of this nature have led governments around the world to ban some of these toys.

To complement recent efforts in analyzing and quantifying security and privacy issues of smart toys, we set out to create two thorough analysis frameworks that are specifically crafted for smart toys. The first framework is designed to analyze legally-binding privacy policies and terms-of-use documentation of smart toys. It is based on a set of privacy-sensitive criteria that we carefully define to systematically evaluate selected privacy aspects of smart toys. We augment our work with a static analysis for the companion Android apps, which are, in most cases, essential for intended functioning of the toys. We use our framework to evaluate a representative set of 11 smart toys, along with 11 companion apps. Our analysis highlights several instances of unnecessary collection of privacy-sensitive information, the use of over-privileged apps, incomplete/lack of information about data storage practices and legal compliance. The proposed framework is a step towards enabling a comparison of smart toys from a privacy perspective, which can be useful to parents, regulatory bodies, and law-makers.

The second framework is used to investigate security and privacy practices - based on experimental analysis - of those specific kinds of IoT devices. In particular, we inspect the real practice of smart toys to determine the personal information they collect and security measures used to protect them. We also investigate potential security and privacy flaws in smart toys that can lead to leakage of private information, or allow an adversary to control the toy to lure, harm, or distress a child. Smart toys pose risks unique to this category of devices, and our work is intended to define these risks and assess a subset of toys against them. We perform a thorough experimental analysis of five smart toys and their companion apps. Our systematic analysis has uncovered that several of these toys may expose children to multiple threats through physical, nearby, or remote access to the toy.

The presented frameworks unite and complement several existing adhoc analyses, and help comprehensive evaluation of other smart toys.

Acknowledgments

I would like to express my gratitude to my supervisor, Dr. Amr Youssef. He has always been caring and encouraging. He showed an excellent guidance and he was always available whenever I needed to discuss anything about my research or even things not related to my research. He also showed amazing leadership skills which made working under his supervision both a pleasure and a great learning experience. I am also grateful to Dr. Mohammad Mannan for his guidance in the world of cyber security. I wish to thank him for his kindness and availability, and for the friendly environment he provided to me.

I would like to thank my lab-mates and Colleagues: Sharon Shasha, Louis Waked, Md Zakir Hossen, Mouna Nakkar, Ahmed Abdelkhalek, Mohamed Nabil, and Abdullah Al-Barakati, for their friendship and for sharing their experiences with me generously. Special thanks to all my instructors at CIISE. This list includes Dr. Jeremy Clark, Dr. Makan Pourzandi, Dr. Lingyu Wang, and Dr. Mohammad Mannan. They all provided me with the opportunity to learn in a positive learning environment and made me more and more interested in all aspects of systems security.

Finally, I would like to thank my family and friends for their respective support and the insights they provided me throughout the journey of my master degree.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Smart toys are gaining popularity in recent times with a rapid growth in sales every year.[1] The apparent educational value has led more parents to adopt these toys for their children. Advances in voice recognition technologies, and the introduction of several hardware sensors have enabled new generations of smart toys to be more intelligent, interactive and dynamic than their predecessors. On the other hand, these enhanced capabilities allow the toys to collect a wide array of personal/device information that could be used for profiling individual children. As many of these toys are also Internet-connected, exposure of the collected information from the toys, personal devices, back-end servers and third-parties, can be a serious threat for the security and privacy of children (including the risk of identity theft; see e.g., [28]).

[1] According to Juniper Research [79], smart toy sales are expected to grow up to three times in a span of just five years from 2017 to 2022.

Recognizing these unique risks to children, governments and regulatory authorities in different regions are introducing specialized laws/acts, e.g., the US Children's Online Privacy Protection Act (COPPA [22]); see also the EU General Data Protection Regulation (GDPR [32]). However, the latitude of privacy concerns for children is getting wider with the revelation of large scale data breaches and loopholes in security mechanisms of the these toys. For example, the VTech leak in 2015 [65] and the recent CloudPets leak in 2017 [48] exposed personal data of nearly six million parents and children; in both cases, adequate security measures were sorely lacking (see also [68]). My Friend Cayla has been banned in Germany for its insecure bluetooth connection [70], allowing a nearby attacker (e.g., up to 15 meters) to interact with children and spy on them. Moreover, the information collection practices in smart toys have also been alarmingly criticized recently [20, 21, 23, 36, 84].

The lack of easily accessible privacy policies and the inherent complexity of these policies hinder the parents' ability to understand potential risks of smart toys and their companion apps for their children. Office of the Privacy Commissioner of Canada (OPC) also offers guidelines on how information related to children can be collected. According to OPC, collection, use and sharing of information about children under 13 years of age, must be approved by their parents. However, in a recent study [59] parents were found to be paying no attention to privacy warnings before allowing their children to play with toys with known security and privacy issues, such as Hello Barbie.

Smart toys are in a position to be continually gathering information about a child, often accompanying a child wherever they go [60]. Many smart toys have onboard sensors and/or actuators, heightening the risks of an exploit [44]. A toy that is continually broadcasting its presence over Bluetooth provides a means of tracking a child through physical space. A toy with sensors, like a microphone or camera, can be made to illicitly transmit sensitive data, like voice or photos, while a toy with actuators, like a motor, can be hijacked to make the toy behave in unintended and potentially harmful ways. For instance, a toy that can be made to move in the direction of the attacker can induce a child to follow the toy directly to the attacker, and a toy with speakers can be made to voice distressing audio [88].

Furthermore, the widespread use of companion apps increases a smart toy's attack surface. An inadequately secured or overprivileged companion app may be exploited to access the smartphone's microphone, camera, or GPS [66], or directly control the toy, and any communication between the toy and the app, if unencrypted, can be intercepted or modified. In addition, many apps make use of third-party ads and analytics servers, which pose their own risks [39]. Children are particularly susceptible to being influenced by advertising [62], especially as they are often unable to distinguish it from other content [67], and parents tend to underestimate the effect that online advertising has on their children [24]. Even apps that do not directly advertise to children may collect analytics data on behalf of third parties, with children more prone to being tracked online than adults [13]. Such tracking info can be highly intrusive [72], allowing an analytics server to build a strong demographic and behavioral profile of the child [58] that can subsequently be used to display targeted ads to the child in unrelated apps and websites [31].

1.2 Contributions

Our contributions can be summarized as follows:

- We propose a comprehensive framework with a set of privacy-sensitive criteria to evaluate privacy policies and terms of use documents of smart toys. We augment the framework with static analysis of companion apps that are essential for the toys' functioning. We evaluate 11 recent smart toys using our framework, and compare them based on the proposed privacy criteria. The framework apparently captures the most serious privacy considerations, and can be used to evaluate a diverse set of smart toys and their companion apps. Using this framework, we show that most toys collect privacy-sensitive information and share them with third parties for unclear purposes. Moreover, companion apps are found to be largely over-privileged, requesting dangerous permissions that are not necessary, or not used at all.

- We develop an experimental framework for evaluating the security and privacy of smart toys. The framework encompasses personally identifiable information (PII) collected and transmitted by smart toys and security measures that have been taken to protect them, as well as PII collected and transmitted to third parties such as ads and analytics services, and the third parties' TLS practices. Our experimental setup uses network traffic analysis, smartphone app reverse engineering, code analysis, and Bluetooth analysis. We also present a carefully defined set of attacks relevant to smart toys, including some that are specific to smart toys. These attacks are categorized based on attacker proximity to the toy (physical, nearby, or remote access).

Using this experimental framework, we investigate a representative set of five smart toys using our experimental setup to expose potential PII leakage, weak security measures, and other vulnerabilities. By utilizing this experimental framework, we uncover excessive collection of unique identifiers that facilitate tracking users across different services or platforms, and sending children's PII to unauthorized entities. Our results also show that several toys expose children to physical, nearby, or remote access threats. We outline a list of best practices smart toys makers can apply to facilitate better security and privacy protection.

Some of the above contributions have been published in [57].

Responsible Disclosure. As part of responsible disclosure, we shared our results with all the toy companies as follows. On Nov. 18, 2017, we have shared our results with all the toy companies in our privacy policies evaluation - through dedicated email addresses for sharing privacy concerns, when available (for four companies); otherwise, we used general support emails or web forms (for seven companies). As of the date of submitting this thesis, Toymail and Sphero have acknowledged the receipt of our report (beyond automated responses). Two emails bounced back with an invalid email address error.

For the experimental analysis, we have contacted the manufacturers of the five toys mentioned in this work and shared our findings. Wiggy Piggy Bank has responded quickly and mentioned that they are looking into the vulnerabilities and possible fixes. Hello Barbie responded that their technical team may get back to us if they have any concerns. We have also received automatic replies from Cloudpets and Toymail; Wowwee Chip has not responded yet.

1.3 Thesis Organization

The rest of the thesis is organized as follows. In Chapter 2, we first present a brief literature review of privacy policies analysis in general and for smart toys in particular. After that, we discuss the security breaches and privacy leakage reported in both the academia and in the industry. In Chapter 3, we present our comprehensive framework for evaluating privacy practices of smart toys based on their legally-binding documentation and in Chapter 4, we present our experimental evaluation framework for smart toys privacy and security practices. Finally, in Chapter 5, we conclude by presenting recommendations and suggestions for best privacy and security practices. We also provide some future research directions.

Chapter 2

Background and Related Work

This chapter covers some necessary background and literature related to this dissertation.

2.1 Background

Smart toys are equipped with sensitive input/output devices including speakers, microphones, cameras, and GPS. Some of these toys have connectivity capabilities including WiFi and Bluetooth. They can also connect to companion mobile apps. In addition, they may also collect children and parents personal information including name, address, date of birth, age, photos, and voice recordings. Smart toys collect these data to present a personalized experience for the children. However, PII collection can be a problem if improper security measures are applied to protect either the locally stored data or the data in the cloud. For example, My Friend Cayla can be hacked due to a vulnerability in the doll software [16] which can lead to personal information leakage. The toy also allows unauthorized Bluetooth connection from any smart phone within 15m from the toy.

In addition, collecting unique identifiers that can uniquely identify children through different services or platforms and sharing these information with third parities can facilitate children tracking and more alarmingly, if these information are stored or shared insecurely, could allow adversaries to track children.

Furthermore, privacy policy is the place where smart toys manufacturers indicate privacy aspects related to their toys. According to COPPA, smart toys manufacturers must state clearly in the privacy policy children personal information practice. According to OPC, privacy policy must indicate clearly, and not in a generic way, the privacy practice of the toy [71].

In fact, the information collection practices in smart toys have been scrutinized recently. For example, the Campaign for a Commercial-Free Childhood (CCFC) condemned the way Hello Barbie collects children's data [36]. The privacy policy of Hello Barbie stated that they may use the collected data "for other research and development and data analysis purposes," without a clearer definition of the scope and extent to which the information can be used. Such vague explanations about data practices may allow them to use the collected information for a wide range of purposes (see e.g., [84]). The US Federal Trade Commission (FTC) has conducted several studies (see e.g., [20, 21, 23]) in recent years, highlighting the lack of disclosure about data practices in mobile apps targeting children and teens. These FTC reports identified an increased availability of privacy policies: 45% apps contained direct links to their privacy policies in the latest survey [23] in comparison to 16-20% in the earliest one [20]; however, ideally, 100% apps should provide a direct privacy policy link.

In this work, we investigate smart toys to determine PII they collect, security measures they apply, and common vulnerabilities they have. We also analyze legally-binding privacy policies and terms of use documentation of smart toys to indicate privacy sensitive features that smart toys manufacturer should take care while they are designing the toys

2.2 Related Work

Below we discuss related work from early research on smart toys security, privacy, and privacy polices analysis.

2.2.1 Security breaches and privacy leakage

Smart electronics for children have made the news for security breaches since it was first discovered that strangers could hack baby monitors to view children sleeping and even talk to them directly [76]. Like wearables for adults, many smart toys are Bluetooth-enabled and accompany children outside the home, and previous research on fitness trackers has uncovered numerous vulnerabilities such as location tracking [42]. In 2016, the Norwegian Consumer Council issued a report [37] highlighting dangerous security vulnerabilities in 3 toys: i-Que, My Friend Cayla, and Hello Barbie, leading Germany to ban Cayla in February 2017 and brand her an “illegal espionage apparatus” [17].

Several security breaches involving the VTech InnoTab MAX, examined in this work, have been found. Many of the vulnerabilities underlying these attacks have been addressed in subsequent years, but others have not. In 2015, VTech's Learning Lodge database, used by the VTech suite of children's tablets, suffered a severe breach, exposing the PII of 6.3 million children's profiles [49]. The breach was due to a combination of insecure practices. HTTP traffic was not secured using TLS, and personal information was sent in plaintext. The database, containing highly personal information, was improperly secured, and vulnerable to a simple SQL injection attack. Furthermore, account passwords were stored using an MD5 hash. Following the widespread exposure of this breach, covered in many major media outlets and drawing extensive scrutiny, VTech brought all affected services offline pending a security review [89]. Also in 2016, the UK-based security firm Pen Test Partners found that the VTech InnoTab MAX is vulnerable to trivial data extraction [73]. Our results indicate this flaw has not been fixed. Later in 2016, it was found that pornography could be easily accessed through the built-in browser by using Google Translate, formerly whitelisted but since removed by VTech [46]. In February 2016, security firm Rapid7 found severe vulnerabilities in the Fisher Price Smart Monkey, since fixed [77].

Hello Barbie has been extensively scrutinized for its privacy and security practices [34, 47]. Hello Barbie encourages children to divulge intimate details about themselves that do not remain private, but are shared with Mattel's partner, ToyTalk, and with parents through the web portal [52, 85], presenting legal implications in the event that a child discloses physical or sexual abuse to the toy [64]. A study by Somerset Recon Inc. [82] found that Hello Barbie was susceptible to a number of vulnerabilities, some fixed in a timely fashion, but our work finds that others, like broadcasting an open hotspot and allowing unauthorized configuration during pairing, are still present. Popular brands of children's smart watches have come in for scrutiny by the Norwegian Consumers' Council, which issued a report in October 2017 highlighting vulnerabilities in all watches they examined [38]. A month later, in November 2017, "Which? UK" issued a report examining common Bluetooth vulnerabilities in children's toys that allowed them to be taken over by potentially malevolent third parties [91]. Given the stakes at risk, the U.S. Senate tabled a report on smart toy security in December 2016 [69]. The following July, the FBI issued an alert warning parents against privacy and security concerns regarding smart toys [87].

Denning el al. [29] analyzed the security and privacy of three robots, two of which are marketed for children: the Erector SpyKee and Wowwee RoboSapien V2. The Spykee features a camera, mic, and speaker, and is controlled through companion software running on a PC over WiFi. The study showed login credentials are transmitted in plaintext and video is streamed unencrypted, allowing even a passive eavesdropper to sniff the login credentials and control the robot over home WiFi and remotely over the Internet. The authors also present psychological attacks in which an adversary could coerce the robot to generate audio that is distressing to children. The severity of an attack can be amplified by using multiple robots together. The authors manipulated the RoboSapien V2 to move a set of house keys within another robot's camera field of view as it streamed video, exposing the house keys to physical duplication.

Rafferty at al. [75] argue that the traditional access control model is insufficient to protect the privacy of children using smart toys. They introduce a conceptual model for smart toys that allows a parent to configure privacy rules and receive notifications about sensitive data disclosure. The model assumes that children are oblivious to privacy concerns and as a result cannot adequately protect themselves online or anticipate the consequences of leaking private information to smart toy makers and other third parties.

2.2.2 Privacy policies analysis

Natural language privacy policies have long been the standard form of notification to users about privacy implications of a service. However, their length and complexity put extra burden on the users' who rarely read and understand them. Identifying the best format to represent privacy implications is non-trivial. Earlier work in this domain has largely been focused on proposing alternatives, or quantifying data practices from legacy privacy policies. Several studies (e.g., [12, 81]) utilize Natural Language Processing (NLP) techniques to identify important data collection practices from privacy policies. Sadeh et al. [81] leverage advances in NLP and machine learning techniques in combination with crowd-sourcing to extract key privacy information from privacy policies, and then present the policies in a user-friendly manner.

Costante et al. [25] propose a solution using Information Extraction (IE) techniques to analyze website privacy policies regarding what data about a visitor is collected. Zimmeck et al. [95] propose a system to automatically examine compliance of Android apps with their privacy policies, by performing static analysis of Android apps and extracting privacy information from their policies. They found that 71% of the apps that do not provide a privacy policy, but collect at least one PII item, and the ones with privacy policies show significant inconsistencies between policies and actual app (code) behavior. Our work is based on careful manual analysis of available privacy policy and terms of use documentation as the number of toys/apps we analyze is limited.

Earlier research analyzed privacy policies to infer data practices in more generic cases, without taking the target user base into consideration. The Explore Privacy Policies project [74] highlights privacy practices of websites. Hoke et al. [43] study privacy policies of 75 tracking companies to examine compliance with self-regulatory guidelines. Costante et al. [26] assess the completeness of privacy policies by comparing them against a set of privacy categories. Cranor et al. [27] utilize web crawling and document parsing to analyze *model* privacy forms of a large number of financial institutions, and found many instances where users' right to control the sharing of information was violated. Various studies focus on privacy of Android apps in general. Kong et al. [54] propose a system called AUTOREB that maps reviews of Android apps to security and privacy behaviors. Zhang et al. [94] attempt to generate security related app description by analyzing the app code.

Another line of work explores the design of user-friendly privacy policy interfaces and formats that would facilitate users' understanding of data practices (e.g., [45, 53, 78, 90]). Kelley et al. [53] develop a solution inspired by nutrition labels that represents the information collection practices in a grid view. Holtz et al. [45] propose the use of privacy icons in addition to the written policies to express data practices in a more effective way.

Several studies have found serious privacy and security issues in connected toys (see e.g., [30, 83]). Security analysis of Hello Barbie reveals several loopholes in its security mechanism [83], including an unencrypted WiFi network used to configure the toy. McReynolds et al. [59] study the expectations and concerns of both parents and children regarding the use of connected smart toys. A report from Future Privacy Forum [40] explores privacy concerns related to microphone-enabled devices, including smart toys such as Hello Barbie, and suggests best practices for devices equipped with microphone. Yankson et al. [93] discuss privacy implications of connected smart toys and propose some best practices that could be embraced by both parents and toy companies.

Chapter 3

A Comprehensive Analytical Framework for Evaluating Smart Toys' Privacy Practices

In this chapter, we propose a broad range of criteria to analyze various privacy aspects of smart toys. In particular, we define a set of privacy-sensitive features as part of our analysis framework. We evaluate a representative set of 11 smart toys and their companion apps; Table 1 lists our selected toys and the hardware sensors and communication channels they are equipped with (for more info on the toys, see Appendix A and Appendix B). Furthermore, to verify permission usage and identify information collection and potential misbehaviors, we perform static analysis of the companion apps.

Table 1: Available sensors and communication channels in our selected smart toys

	Devices and Sensors									Comm.		
	Microphone	Speaker	Camera	IR-Vision	Gyroscope	Motion Detector	Touch Detector	Accelerometer	Thermometer	WiFi	Bluetooth	IR
Hello Barbie	✓	✓								✓		
Toymail	✓	✓								✓		
Sphero BB-8	✓	✓			✓	✓				✓	✓	
Wowwee Chip	✓	✓		✓	✓	✓	✓	✓		✓	✓	✓
Smart Toy Monkey	✓	✓								✓	✓	
CogniToys Dino	✓	✓								✓	✓	
Edwin the Duck		✓							✓	✓	✓	
Anki Cozmo		✓	✓		✓					✓	✓	
My Friend Cayla	✓	✓								✓	✓	
I-Que Robot	✓	✓								✓	✓	
Zenbo	✓	✓	✓		✓	✓				✓	✓	

3.1 Evaluation Framework

In this section, we define a diverse set of criteria that we believe are representative of various privacy aspects in smart toys. The criteria we explore encompass five categories: application authenticity and permissions; privacy policy documentation; ToU documentation; information collection; and information storage, sharing and protection. The selected set of criteria are initially inspired by the data quantification categories of Sadeh et al. [81], and iteratively refined during our analysis of the smart toys. To augment our analysis of the available documentation, we perform static analysis of the companion apps. We use the word *feature* and *criterion* interchangeably throughout the rest of the chapter.

3.1.1 Criteria

In this section, we define the rating criteria of our evaluation framework. For each criterion, a toy may fully or partially satisfy it, not satisfy it, or may not provide relevant information.

3.1.1.1 Application Authenticity and Permissions

Smart toys are generally accompanied by a companion mobile app, which can be downloaded from an app market (e.g., Google Play). Downloading the right app affiliated with the toy is essential for its intended functioning, and for security and privacy reasons (e.g., to avoid downloading a repackaged app with malware/adware/spyware). Moreover, over-privileged apps can pose privacy risks, as they may request extraneous permissions, which can be used to access sensitive information. We define the following criteria to cover these concerns.

A1 App-website-links: The official website of the toy contains a link to download its companion app from an app market (e.g., Google Play), and the app contains a link to the official toy website. This bi-directional linking verifies the app's origin; a toy is partially granted this feature if one of them is missing.

A2 Reasonable-permissions: The companion app requests only for permissions that are necessary for its intended functionality. We perform static analysis of the companion apps together with manual evaluation of provided features to rate this criterion.

3.1.1.2 Privacy Policy Documentation

Privacy policy communicates data practices of a service. An easily accessible and up to date privacy policy is essential for communicating privacy implications to parents who are responsible for permitting the collection of their children's information.

P1 Store-app-website-links: The companion app, its Google Play page, and the official website contain a link to the toy privacy policy; partially granted if either of them is missing.

P2 Update-info-notification: Any changes in the privacy policy should be reported to users. To fully satisfy this feature, the toy must have the date of last update mentioned in the policy along with explanation of how they report the updates to users; partially granted if either of them is missing.

3.1.1.3 Terms of Use Documentation

We consider ToU as an important aspect in our framework as it may contain important privacy practices; we define the following two features.

T1 Store-app-website-links: Similar to P1 (for ToU).

T2 Update-info-notification: Similar to P2 (for ToU).

3.1.1.4 Information Collection

Smart toys can collect a wide range of information during setup/installation (companion app), or while being used by children. They must comply with privacy acts or laws (specific to children or in general) of different regions where the toys are sold. We define the following criteria to highlight data collection practices in smart toys.

C1 Laws/acts-compliant: The privacy policy of a toy states explicitly the laws/acts they comply with, and the jurisdiction(s) under which they operate.

C2 Reasonable-PII-collection: The toy and its companion app collects *reasonable* PII. We define email address as reasonable PII, assuming email is used to communicate privacy policy and ToU changes to the user. Any PII beyond email is considered unreasonable.

C3 No-website-data-collection: The toy's website does not collect any information that can be used for user tracking or serving personalized ads.

3.1.1.5 Information Storage, Sharing and Protection

Recent data breaches (e.g., [48, 65]) raise questions on secure data storage and protection practices of smart toys. This is a major concern as one of the leaks ([65]) contained information that could lead to identification of individual children and their location. We define eight features under this category.

S1 Data-storage-location: The location of data storage is stated in the toy's privacy policy. Based on their storage location, companies may be subjected to specific regulations in case of a data breach incident. For example, data breach regulations in US states differ;[1] see also EU e-Privacy Directive[2] and GDPR [32].

S2 No-third-party-PII-sharing: Any information collected through the toy is not shared with third parties.

S3 Parental-PII-control: Parents can permanently delete the information collected by the

[1] See: http://www.ncsl.org/research/telecommunications-and-information-technology/security-breach-notification-laws.aspx

[2] http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0058:en:HTML

toy and its companion app.

S4 PII-protection: The measures taken by the toy manufacturer to protect the collected information is properly documented in the privacy policy or the ToU. Currently, we simply rate a toy based on its use of TLS for all communications between the toy/user/device and back-end servers.

S5 Dedicated-privacy-support: The toy manufacturer offers dedicated support for privacy concerns (e.g., via a specific web page or email address, instead of a generic support contact).

S6 Protection-program-participant: The toy manufacturer participates in independent programs that provide additional support to users to resolve privacy issues. Such programs may include Judicial Arbitration and Mediation Services (JAMS [50]) and TRUSTe [86].

S7 Bug-bounty-participant: The toy manufacturer participates in bug bounty programs that encourage people to identify security and privacy issues in their toys/apps. Such participation may indicate a strong commitment towards information security and protection.

S8 Do-not-track-support: The documentation explains how the toy's website handles Do Not Track (DNT) requests. A DNT request means the user does not want his browsing data to be collected and tracked across sessions/devices.

3.1.2 Static Analysis

We use two complementary tools for static analysis: RiskInDroid [61] and Androwarn [56]. We use RiskInDroid to analyze the permission usage of companion apps in order to identify over-privileges (we limit our app analysis to Android apps only). RiskInDroid uses machine learning techniques to quantify risks posed by Android apps, and assigns a risk value between 0 to 100; higher value indicates higher risks. It uses static analysis to infer permission utilization in the app code, and categorizes them in four sets. In our analysis of companion apps, we focus on two sets, namely the declared permissions in the app Manifest, and the permissions that are actually used in the code. This is done by extracting the API calls from the decompiled source code and mapping them to the required permissions (using PScout [14]).

Table 2: Comparative evaluation of the representative smart toys

Product	A1: App-website-links	A2: Reasonable-permissions	P1: Store-app-website-links	P2: Update-info-notification	T1: Store-app-website-links	T2: Update-info-notifcation	C1: Legal-compliant	C2: Reasonable-PII-collection	C3: No-website-data-collection	S1: Data-storage-location	S2: No-third-party-PII-sharing	S3: Parental-PII-control	S4: PII-protection	S5: Dedicated-privacy-support	S6: Protection-program-participant	S7: Bug-bounty-participant	S8: Do-not-track-support
Hello Barbie	●		●	●	●	●	●					●	●	●	●	●	N/A
Toymail	●		○	○	●	○	●					●	●				N/A
Sphero BB-8	●		●	○	●	○	●						●				○
Wowwee Chip	●		○		N/A	N/A	●				N/A	○	N/A				N/A
Smart Toy Monkey	●		●	○	○	○	●					●	N/A	●			N/A
CogniToys Dino	○		○	○	●	○	●					●	○				N/A
Edwin the Duck	○		○	○	●		●					●	○				N/A
Anki Cozmo	○		●	●	○	○	●	N/A		●		●	●	●	●		
My Friend Cayla	●		○	○	N/A	N/A	●						●				N/A
I-Que Robot	●		●	○	●	○	●						●				N/A
Zenbo	○		○	○	●	○							●	●			N/A

● = offers the feature; ○ = Partially offers the feature; no circle = does not offer the feature; N/A = information unavailable.

We denote over-privilege as having permissions in the Manifest that are not utilized in the app code. Such over-privilege does not necessarily imply hidden intents, as there could be several benign reasons (e.g., developer mistake). Moreover, in newer versions of Android (6.0 and later), the app will not be granted the dangerous permissions during installation, rather it would ask for permissions during runtime and the user may choose to grant or deny them. However, the app may still ask the users for permissions that they do not need or use at the moment, but once granted can utilize them in later versions of the app.

We use another static analysis tool Androwarn [56] to identify potential misbehaviors and information collection. In contrast to RiskInDroid, Androwarn uses a combination of structural and data-flow analysis to identify suspicious behaviors including exfiltration of sensitive information (e.g., device unique identifiers and geolocation via GPS/WiFi), and abuse of functionality (e.g., making phone calls, sending SMSes, and recording audio/video).

We choose RiskInDroid as it is one of the most recent tools of its kind (published in 2017). In contrast, Androwarn is an open source tool available since 2013 (used in other app analysis studies). The accuracy of the results of our analysis is tied to the tools we use. We encountered only one mismatch in outcomes from the tools (for Toymail, RiskInDroid labels READ_CONTACTS as unused, but Androwarn's data flow analysis finds its use for reading the contact list); we acknowledge the fact that using different tools may yield somewhat different results.

3.2 Analysis and Results

We now use our criteria to evaluate a representative set of 11 smart toys. We manually examine the privacy policies and ToU documentation to check whether the proposed privacy criteria are fulfilled by the toys or not (between June 2017 to July 2017, see Table 10 in the

Appendix for links to the documents). We also statically analyze the companion apps to identify over-privileged apps, personal/device identifier leakage, and suspicious behaviors. Below, we elaborate how each toy is rated in our framework; note that, for brevity, we discuss a feature in the text if it requires some explanation. For a quick summary of our results and an overall picture, see Table 2 and Section 3.3. Table 3 list permissions declared in the app Manifests, and Figure 1 shows permission utilization. Tables 4, 5, and 6 summarize PII collection, device information collection, and toy/app usage collection, respectively.

Hello Barbie. Hello Barbie's companion app does not satisfy *Reasonable-permissions* as it declares seven permissions in the Manifest but uses only four. The unused permissions include write access to the internal storage and read/write access to the external storage. Hello Barbie provides *Store-app-website-links* to the privacy policy, and satisfies *Update-info-notification*. It also mentions that in some cases, it obtains the user's prior verifiable approval before updating the policy (no explicit mention of approval mechanisms).

Hello Barbie is *Legal-compliant* with COPPA. It does not satisfy *Reasonable-PII-collection*: from the user, it collects email address, voice recordings, and child birthday (Table 4); from the device, it collects device model and name, IP address, operating system, browser type, mobile network information (Table 5). It also collects service usage information including information about how the app features and speech processing services are used, and information about the number, frequency and length of each session (Table 6). In addition, the toy does not satisfy *No-website-data-collection* as it sets cookies and web beacons.

Hello Barbie does not satisfy *No-third-party-PII-sharing* as it shares personal information with vendors, consultants, and other service providers. Hello Barbie provides *Parental-PII-control* through the parent's account. The toy also provides *PII-protection* through secure encrypted data transmission (TLS). In addition, the child's voice recordings are not stored locally on the toy (WiFi credentials are stored). However, a recent study [83] shows that Hello Barbie suffers from many vulnerabilities including using weak passwords, no password brute force protection, using unencrypted WiFi network to configure the toy, and not requiring unique authentication to modify the configuration of the toy. As our analysis is based on the information obtained from the documentation, we grant *PII-protection* to Hello Barbie. We follow the same principle for other toys, assuming any reported vulnerabilities will be promptly fixed.

Hello Barbie is *Protection-program-participant*: the user can directly submit a complaint to JAMS to resolve a dispute. In addition, Hello Barbie states that its server, ToyTalk, is subject to the investigatory and enforcement powers of the US FTC. It is *Bug-bounty-participant* according to HackerOne [41]; however, Hello Barbie mentions in the ToU document that attempts of reverse engineering, decompiling, or discovering the source code are disallowed. The toy does not clearly state the *Data-storage-location*: it states that the location can be in the USA or other countries.

Toymail. Toymail does not satisfy *Reasonable-permissions* as it declares 14 permissions in the Manifest but does not use six, including get accounts, access the camera and read/write external storage. In addition, our static analysis shows that the companion app can abuse the telephony service to make phone calls without user consent, and it can read and edit

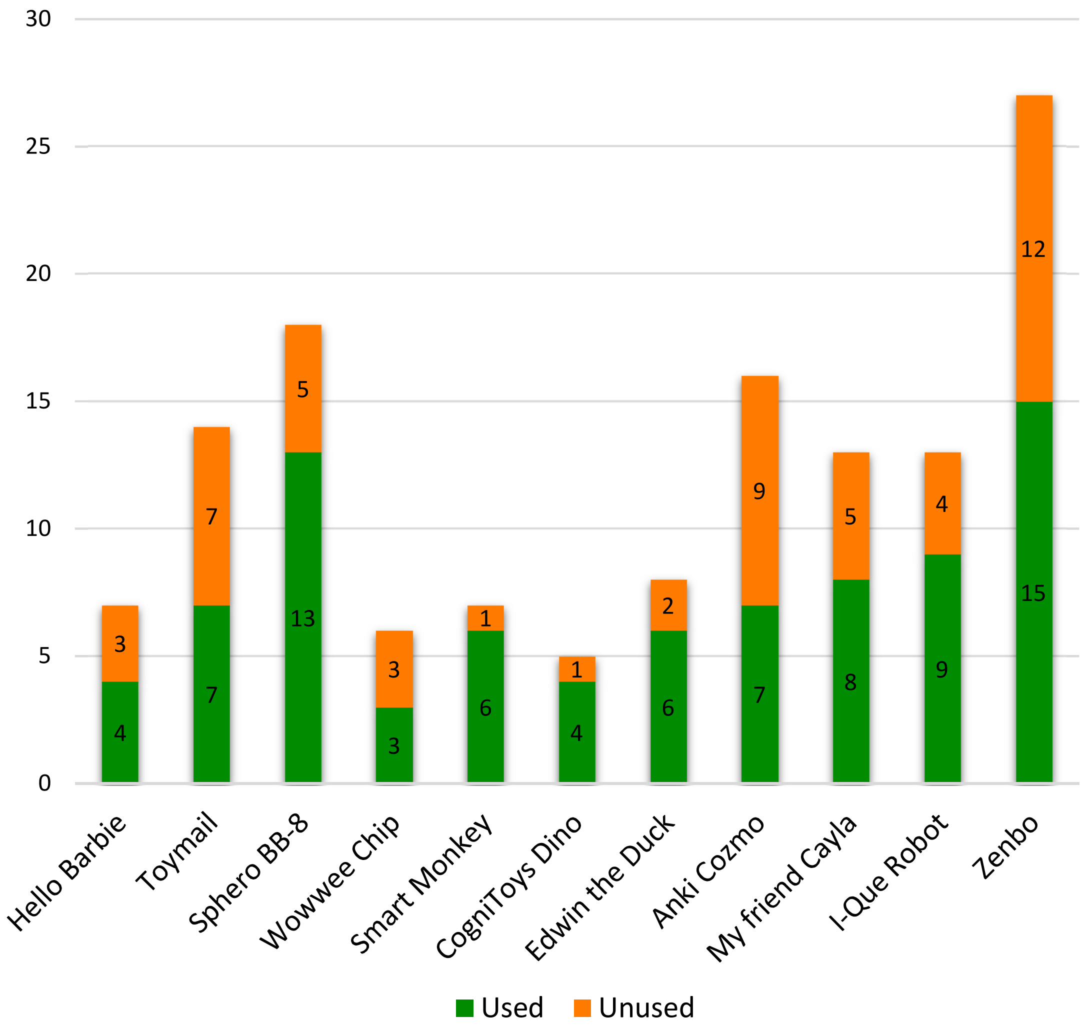


Figure 1: Permissions requested (used vs. unused) by the companion apps

Table 3: Permissions requested by companion apps

	Data Access			Communication						Functionalities													Control						
Product	Photos, Media, Files and Storage	USB storage	Contacts	View WiFi connections	Full network access	WiFi multicast reception	Control WiFi	Pair with bluetooth devices	Access bluetooth settings	Take pictures and videos	Record audio	Receive data from Internet	Approximate location	Precise location	Read phone status and identity	Use accounts on the device	Google Play license check	Send sticky broadcast	Retrieve running apps	Disable screen lock	Control flashlight	Run at startup	Audio settings	Display settings	Control vibration	Prevent device from sleeping	Modify system settings	Control accounts	Create accounts
Hello Barbie	✓	✓		✓	✓																								
Toymail		✓	✓	✓	✓					✓	✓	✓													✓	✓	✓		
Sphero BB-8	✓	✓		✓	✓			✓	✓	✓	✓	✓	✓		✓	✓	✓								✓	✓			
Wowwee Chip	✓	✓			✓			✓	✓																	✓			
Smart Toy Monkey				✓	✓		✓	✓	✓				✓																
CogniToys Dino				✓	✓									✓												✓			
Edwin the Duck	✓	✓			✓			✓	✓				✓	✓									✓						
Anki Cozmo	✓	✓		✓	✓	✓	✓	✓	✓			✓	✓				✓								✓	✓			
My Friend Cayla	✓	✓		✓	✓			✓	✓		✓						✓	✓					✓	✓		✓			
I-Que Robot	✓	✓		✓	✓			✓	✓		✓						✓	✓					✓	✓		✓			
Zenbo	✓	✓		✓	✓		✓	✓	✓	✓	✓	✓			✓	✓			✓	✓	✓	✓	✓		✓	✓	✓	✓	✓

contacts.

Toymail partially satisfies *Store-app-website-links* (P1); it includes some privacy information in the ToU document, including collection of personal information (e.g., voice recordings), service usage information, and sharing policy of personal information. It also partially satisfies *Update-info-notification* (P2) since it fails to notify the user with any update details to the privacy policy, although it indicates the update date (similar rating for ToU T2).

Toymail is *Legal-compliant* with COPPA, and the federal courts in Michigan have exclusive jurisdiction. It does not satisfy *Reasonable-PII-collection* as it collects email address, child's name, image, and birthday, time zone, sound bite of the child's name (see Tables 4, 5, and 6). Moreover, our data flow analysis for the companion app reveals that it also collects the unique device ID that can be used to fingerprint the user's device and can allow tracking the user across different services [55]. In addition, it collects information from users when they use the toy's website, including IP address, browser type and version,

Table 4: Personal information collected by smart toys via the toys and companion apps

Product	Name (parent)	Name (child)	Gender	Physical address	Country	Postal code	Email address	Social media profile	Telephone number	Child image	Voice recordings	Child birthday	Child's interests	Payment info	Demographic info
Hello Barbie							✓				✓	✓			
Toymail		✓					✓			✓		✓			
Sphero BB-8	✓		✓	✓			✓							✓	✓
Smart Toy Monkey	✓	✓					✓		✓						✓
CogniToys Dino	✓	✓	✓	✓			✓		✓			✓	✓	✓	
Edwin the Duck	✓			✓			✓		✓					✓	
My Friend Cayla	✓					✓	✓				✓	✓			
I-Que Robot	✓					✓	✓				✓	✓			
Zenbo	✓		✓	✓	✓	✓	✓	✓						✓	

and cookies. It also uses Google analytics service, hence it does not satisfy *No-website-data-collection*.

Toymail does not satisfy *Data-storage-location* as it states that data can be stored in the USA or other countries. It does not satisfy *No-third-party-PII-sharing* as it shares personal information with "Electric Imp", service technicians, other Toymail employees, and provides users' personal information in case of a law or court order, or if third-party entities audit its system for security vulnerabilities. Toymail satisfies *PII-protection* (uses TLS). It uses Amazon cloud service to store personal data, and thus it depends on Amazon's security measures to prevent the unauthorized access to its data. Toymail also performs automated deployments and security upgrades, and it uses independent third-party services to audit the system for security vulnerabilities. It is not *Bug-bounty-participant* and prohibits any attempts for reverse engineering, decompiling, or discovering the source code.

Table 5: Device information collected by smart toys (from user devices e.g., smartphone)

Product	Device Unique ID	Serial Number	Mac address	IP address	Device type	Device name and model	Device activation time	WiFi SSID and password	Operating system	Browser type	Device carrier	Internet service provider	Network status	WiFi nearby APs	GPS info
Hello Barbie	×			✓		✓			✓	✓	✓		×		
Toymail	×			✓				✓					×		
Sphero BB-8	×			✓	×	✓			✓	✓	✓	✓			
Wowwee Chip	×										×				
Smart Toy Monkey						✓			✓	✓	✓		×		
CogniToys Dino	×										×		×		
Edwin the Duck	✓			✓	✓				✓	✓	×		×		
Anki Cozmo					×								×		
My Friend Cayla				✓	×						×		×		
I-Que Robot				✓							×		×		
Zenbo	✓	✓	✓	✓	×	✓	✓		✓		✓		✓	✓	✓

✓ refers to information collected by the companion apps as declared in the privacy policy
× refers to information collected by the companion apps but not declared in the privacy policy
blank means no information is collected

Sphero BB-8. Sphero BB-8 does not satisfy *Reasonable-permissions* as it declares 18 permissions but uses 13. It requires location access, which is not necessary for the toy; it requires access to device status and identity, allowing it to collect user's personal information including phone number and device ID. Moreover, the static analysis for the companion app reveals that it can make phone calls without the user's consent.

Sphero BB-8 partially satisfies *Update-info-notification* for ToU (T2) as the toy only notifies users with 'important' changes in the ToU without defining what is considered as important.

Sphero BB-8 is *Legal-compliant* with EU safe harbor agreement [33], which is an agreement between the European Union and USA to protect user information; it is also governed by the laws and courts of Colorado (USA). On the other hand, Sphero BB-8 does not satisfy *Reasonable-PII-collection* as the toy collects excessive information from users; see Tables 4, 5, and 6. Moreover, our static analysis reveals that the toy also collects the unique device ID. Sphero BB-8 does not satisfy *No-website-data-collection* as it gathers information including browser/OS type, IP addresses, referring URL, date/time stamps for the visits, cookies, web beacons. It also indicates that third parities including advertisers, ad measurement services, and ad networks may collect device information and information about the users' online activities over time and across different websites.

Sphero BB-8 does not satisfy *Data-storage-location* as it may store the information in the USA or any other country. It lacks *No-third-party-PII-sharing*: it shares personal information with third-party services who may work for the company, and in the case of corporate restructuring or court order. The toy lacks *Parental-PII-control* as it does not provide an explicit facility for PII-deletion, though it provides an email address, which parents can use to contact the company for deleting personal information; however, it mentions that the information will be deleted from their active server but not from the archive servers. It provides *PII-protection* as it uses encrypted traffic and it hires third-party security experts to audit the network infrastructure. It partially provides *Do-not-track-support* feature because Sphero BB-8 discusses about DNT in their documentation but acknowledges that they may not be able to recognize such requests.

Wowwee Chip. Wowwee Chip lacks *Reasonable-permissions* as only three out of the six requested permissions are used. Wowwee Chip partially provides *Store-app-website-links* (P1) to the privacy policy as there is no link from Google Play to the privacy policy. Wowwee Chip does not provide ToU document, hence *Store-app-website-links* (T1) and *Update-info-notification* (T2) features are inapplicable.

Wowwee Chip is *Legal-compliant* with COPPA. It collects personal information from users when they sign up for email newsletters, or register a product; however, Wowwee Chip does not mention exactly which personal information it collects. Our static analysis reveals that the companion app collects the unique device ID, and thus not satisfying *Reasonable-PII-collection*. Wowwee Chip sets cookies through its website, hence it does not satisfy *No-website-data-collection*.

Wowwee Chip fails to achieve all the features of *Information Storage, Sharing and Protection* except *Parental-PII-control* (partial). Wowwee Chip mentions that users can delete their accounts, though it does not mention whether or not the personal information

Table 6: Toy usage information collected by smart toy companies

Product	Toy's ID	Firmware version	App install and uninstall time	Toy features used	Toy features usage frequency	Session length	Crash history
Hello Barbie				✓	✓	✓	
Toymail	✓	✓					
Sphero BB-8		✓		✓	✓	✓	
Smart Toy Monkey				✓	✓	✓	
My Friend Cayla			✓				✓
I-Que Robot			✓				✓
Zenbo				✓	✓	✓	✓

will be permanently deleted.

Smart Toy Monkey. Smart Toy Monkey lacks *Reasonable-permissions* as it requests location permission in the Manifest, which remains unused in the app. It partially satisfies *Update-info-notification* (P1) as it does not include the last update date in the privacy policy, and states that it may notify users about important changes. Smart Toy Monkey partially satisfies *Store-app-website-links* (T1) for ToU because in Google Play, Smart Toy Monkey mistakenly names the ToU as privacy policy, which can mislead users, although it provides a direct link to the ToU in the companion app and the website. It also partially achieves *Update-info-notification* (T2) because it does not indicate the last update date in the ToU (but notifies users).

Smart Toy Monkey is *Legal-compliant* with US laws, implying COPPA compliance.

It lacks *Reasonable-PII-collection* as it collects unique device ID (revealed by the static analysis). Smart Toy Monkey also states that it optionally collects names, email, telephone number, demographic and other personal information. It does not satisfy *No-website-data-collection* (uses cookies and web beacons).

Smart Toy Monkey does not provide *No-third-party-PII-sharing*: it may use third parties to provide analytics and advertising services, implying that it can share personal information with them. It does not clearly state the *Data-storage-location* as it may store information in the USA or any other country. It is not *Bug-bounty-participant* and it states clearly that it prohibits any attempts for reverse engineering, decompiling, or discovering the source code.

CogniToys Dino. CogniToys Dino partially achieves *App-Website-links*, as it does not provide a link from website to the app. It lacks *Reasonable-permissions* as the companion app requires location access permission (also another unused permission). CogniToys Dino partially satisfies *Store-app-website-links* (P1) as it has no direct link to the privacy policy from the Google Playpage; the policy is reachable through its app interface. It partially satisfies *Update-info-notification* (P2) because it does not notify users with updates in the privacy policy. For ToU, CogniToys Dino partially provides *Update-info-notification* (T2) because it states that users must check back the ToU for any changes, implying that CogniToys Dino does not notify the user with changes.

CogniToys Dino is *Legal-compliant* with US laws, implying COPPA compliance. It lacks *Reasonable-PII-collection* as it collects name, address, mobile phone number, email address, payment information, and child's name, date of birth, and gender. Static analysis reveals that the companion app also collects unique device ID. It does not satisfy *No-website-data-collection* as it collects information about web browser, OS, ISP, IP addresses, device type, viewed pages, the time and duration of visits to the site, and it sets cookies uses Google Analytics services.

CogniToys Dino lacks *Data-storage-location*: it can store user information in the USA or any other country. It does not satisfy *No-third-party-PII-sharing* because it shares information with third-party service providers who work for the company. It also shares personal information in case of corporate restructuring, or in case of a law or court order. We rate CogniToys Dino as partially providing *PII-protection* since it does not mention exactly which kind of measures it takes to protect data, although CogniToys Dino states that it takes physical, electronic, and procedural safeguards to protect the information. It is not a *Bug-bounty-participant*, and prohibits any attempts for reverse engineering, decompiling, or discovering the source code.

Edwin the Duck. Edwin the Duck is labeled as partially satisfying *App-website-links* as its companion app contains a link to its official website but the link from the website to its Google Play page is missing. Static analysis of the app reveals two unused permissions (read/write external storage), making it over-privileged and not satisfying *Reasonable-permissions*. In spite of containing links to the privacy policy from its website, companion app and Google Play page, Edwin the Duck is partially granted *Store-app-website-links* (P1) as the later two link to different documents. *Update-info-notification* (P2) is partially satisfied as both versions of the privacy policy state the last date of update but fail to mention whether they will keep the users updated with any changes in the policy. For ToU, it lacks *Update-info-notification* (T2).

Edwin the Duck does not satisfy *Reasonable-PII-collection* because it collects name, address, email address, type of device, device unique ID, IP address, OS, and browser information. It does not satisfy *No-website-data-collection* as it sets cookies, and collects name, email address, mailing address, phone number, and credit card information.

Edwin the Duck lacks *No-third-party-PII-sharing* as it shares information with third-parties who assist the company in operating and developing the service. It states that non-personally identifiable visitor information may be provided to other parties for marketing, advertising, or for other uses. We rate Edwin the Duck as providing *Parental-PII-control* as it allows parents to delete PII by contacting the company through "Contact us" link in the website. It partially provides *PII-protection*: Edwin the Duck mentions that all sensitive information is transmitted over SSL (but does not clarify what is considered as sensitive, except credit card numbers); at the server-side, Edwin the Duck does not store a user's private information such as credit card and financial information. Edwin the Duck is not *Bug-bounty-participant*, and it prohibits any attempts for reverse engineering, decompiling, or discovering the source code.

Anki Cozmo. Anki Cozmo partly achieves *App-website-links* as the link to the Google Play page of the companion app from its website is unavailable. It lacks *Reasonable-permissions*: the app uses 7 out of 16 requested permissions. The unused permissions include read/write access to the external storage of the device, access to bluetooth settings.

We grant the toy a partial *Store-app-website-links* as the ToU is unavailable in its Google Play page. The toy is granted a partial *Update-info-notification* as there is no mention of how they will report updates in ToU to users. *Reasonable-PII-collection* is not granted as there is no clear explanation of what data they collect from users. *No-third-party-PII-sharing* is not granted as it shares collected information with third party ad networks. The toy achieves *Protection-program-participant* as users can contact TRUSTe [86] and JAMS in case of a dispute.

My Friend Cayla. My Friend Cayla fails to satisfy *Reasonable-permissions*: 5 out of 13 requested permissions are unused, including read/write access to external storage. Moreover, static analysis shows that the app can make phone calls without users' consent. We could not find a link to the privacy policy from the app, therefore, granting a partial *Store-app-website-links*. There is no mention of how it will notify users of any policy changes, although, it contains the last date of update in the privacy policy, thus partially fulfilling *Update-info-notification* (P2).

The toy does not offer *Reasonable-PII-collection* as it collects IP address, zip code, date of birth and voice messages. Furthermore, the static analysis for the companion app shows that it collects the unique device ID. The toy fails to satisfy *No-third-party-PII-sharing* as it shares data with partner organizations. It lacks *Parental-PII-control* as according to its privacy policy, there is some information that cannot be removed completely.

I-Que Robot. I-Que Robot does not satisfy *Reasonable-permissions* as four out of thirteen declared permissions remain unused, including write access to the external storage. Data flow analysis for the app also shows that it can make phone calls. Although the last date of update is mentioned, there is no specific statement in the privacy policy on notifying users about changes in the document, thus partly satisfying *Update-info-notification* (P2); it is similarly rated for T2 (ToU). I-Que Robot collects the same PII as My Friend Cayla, and thus lacks *Reasonable-PII-collection*. The toy shares information with other parties and fails to satisfy *No-third-party-PII-sharing*. The toy uses firewalls and secure databases to achieve *PII-protection*.

Zenbo. Zenbo is partially granted *App-website-links* as the official website does not contain any link to the companion app's Google Play page. The app is also highly over-privileged; it requests 27 permissions, but uses only 15 (unused permissions include: flashlight, recording audio, using camera and modifying system settings). The app requires the permission to retrieve running apps information and to manage accounts on the device. Static analysis also reveals that the companion app can be involved in telephony service abuse by making phone calls without users' consent. Thus Zenbo is not granted *Reasonable-permissions*.

The toy partially achieves *Store-app-website-links* because its Google Play page provides a link to the Mandarin version of the privacy policy and there is no way to switch to the English version, though the website and the app provide links to the English version. The toy is also partially granted *Update-info-notification*: it will notify users if there is any important/big update in the policy. *Update-info-notification* is achieved partly as the procedure of notifying users is not mentioned. We do not grant *Reasonable-PII-collection*: it collects a wide range of PII including name, email, gender and date of birth, if the user decides to login using social media profiles; otherwise, it only requires email address and country.

3.3 Summary of Results

None of the toy companion apps achieve *Reasonable-permissions*. Our static analysis reveals different levels of over-privileges (i.e., more permissions declared in the Manifest than the app needed or used). For example, Smart Toy Monkey and CogniToys Dino declare only one unused permission, but Zenbo has 12 unused permissions. Among the declared and used permissions, there are multiple instances where the requested permissions are not necessary for the toys' intended functioning. For example, Sphero BB-8 requires access to the approximate location; Zenbo requests permissions to allow managing users' accounts on the device. Static analysis also shows that some toys may perform unwanted/suspicious activities surreptitiously. For example, My friend Cayla, I-Que Robot, and Zenbo companion apps can make phone calls without user consent.

Most toys perform poorly in *Information Collection* features. All except Anki Cozmo collect PII that appear unreasonable, and Anki Cozmo fails to declare which PII it collects. Collected PII includes: email address, voice recordings, address, phone number, child's name, image and birthday; device information including device model and name, IP address, OS/browser version, and mobile network information. Collected service usage information includes: information about how the features of the app and speech processing services are used, and information about the number, frequency and length of each session. Moreover, static analysis reveals that some toys may collect personal information that is not mentioned in their privacy policies. For example, Toymail, Sphero BB-8, Wowwee Chip, CogniToys Dino companion apps collect the unique device ID (e.g., IMEI) that can be used to fingerprint the user's device and can allow tracking the user across different services. All the toys fail to achieve *No-website-data-collection* as they at least set cookies and web beacons (which can be used for tracking and serving targeted ads; see e.g., [19]).

Except Wowwee Chip all the toys share PII with third parties. Hello Barbie, Toymail, Smart Toy Monkey, CogniToys Dino, Edwin the Duck, and Anki Cozmo provide full *Parental-PII-control*. Seven toys claim to take security measures for *PII-protection*, two toys do not state exactly which measures they take to protect PII, and two others do not provide any information. Four toys provide a dedicated webform/email address to contact the company in case of any privacy concern about their toys; two of those are *Protection-program-participant* in TRUSTe or JAMS. Hello Barbie is the only toy that is *Bug-bounty-participant*, which may help discover security and privacy flaws in the toy. None of the toys' websites (except Sphero BB-8 partially) respect DNT.

Responsible Disclosure. As part of responsible disclosure, we shared our results with all the toy companies as follows. On Nov. 18, 2017, we have shared our results with all the toy companies in our privacy policies evaluation - through dedicated email addresses for sharing privacy concerns, when available (for four companies); otherwise, we used general support emails or web forms (for seven companies). As of Nov. 29, 2017, Toymail and Sphero have acknowledged the receipt of our report (beyond automated responses). Two emails bounced back with an invalid email address error.

3.4 Conclusion

In this chapter, we present a comprehensive framework for evaluating privacy practices of smart toys – to help us better understand their policies and to be able to compare them. We use our framework to analyze a representative set of 11 smart toys and their companion apps. We believe it can help evaluate other smart toys in the market (with possible extension and refinement). We found several issues in the privacy practices of these smart toys, especially in regards to PII collection, third-party data sharing, web tracking, and data storage location. We augment our policy analysis by statically analyzing the toys' companion apps to determine over-privileges, sensitive PII collection and suspicious behaviors. We found that all the companion apps are over-privileged and collect unnecessary personal information. Our static analysis provides evidence of potential suspicious activities of the companion apps, such as abusing the telephony service. We believe that our framework can facilitate quick and effective comparison of smart toys privacy practices in future, and be useful to parents, law-makers and toy manufacturers.

Chapter 4

Experimental Evaluation of Privacy and Security Practices in Smart Toys

While in chapter 3, we analyze legally-binding privacy policies and terms of use documentation of smart toys to find privacy sensitive features that smart toys manufacturer should take care while they are designing the toys, in this chapter, we set out to investigate the real security and privacy practices of those smart toys. Here, we inspect the actual practice of the toy by putting it under examination in our carefully-built experimental setup to detect the exact PII the toy or the third parity ads and analytics services collect. In chapter 3, we concluded that smart toys manufacturers should state in the legal documentation the security measures smart toys use to protect collected PII. Here, we scrutinize the real security practice of those smart toys. In addition, in this chapter, we rigorously inspect smart toys against especially-prepared list of vulnerabilities that we believe smart toys can be vulnerable to. We categorized those vulnerabilities according to the attacker proximity to the toy

- namely physical, nearby, or remote access.

To summarize, in this chapter, we introduce an experimental framework that we use to systematically analyze a representative subset of smart toys to determine the personal information they collect and security measures used to protect them. We also investigate potential security and privacy flaws in smart toys that can lead to leakage of private information, or allow an adversary to control the toy to lure, harm, or distress a child. In particular, we consider two broad categories of risk: breach of Personally Identifiable Information (PII) and unauthorized control of the toy.

4.1 Analysis framework

Most smart toys we analyze are intended to be used with companion apps running on smartphones or tablets. All the mobile apps we examine, and many of the toys, communicate with remote hosts. This presents a wide attack surface covering toy, app, hosts, and the communication between each pair of these. We consider each component - toy, app, and hosts - as forming a dyad with each other component, and each dyad as a possible attack vector. For example, a toy establishes a connection with its associated mobile companion app, usually over either Bluetooth or 802.11, and if the toy can be configured to have Internet connectivity, it likewise may establish an Internet connection with one or more web-based hosts. The toy, the app, and the hosts are all attack vectors in this scenario, and so too are the connections formed between the toy and its companion app, as well as the toy and its hosts, as depicted in Figure 2.

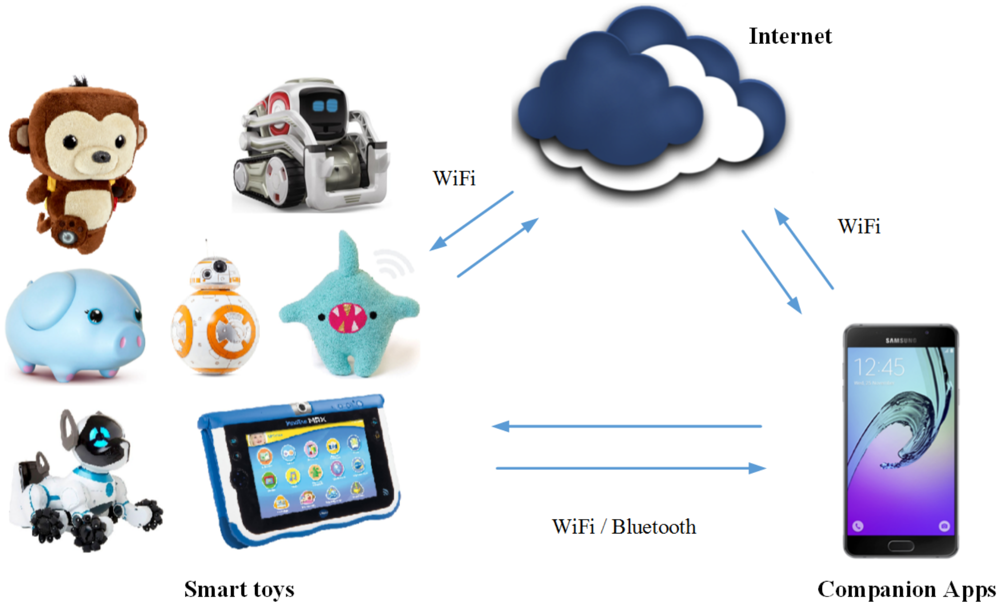


Figure 2: Smart toy attack surface

4.1.1 Investigating PII Collected and Transmitted

PII can be divided into three categories: personal information including name, gender, physical address, email address, telephone number, voice recordings, and photos; device information, including mobile device unique identifiers like IMEI, serial number, MAC address, IP address, and smart toy ID; and service usage information including session start and end time, session duration, and app features. Toys that require the user to create an account typically collect and store more PII than toys that do not. We audit the network traffic of smart toys and their companion apps to determine the PII they collect, transmit, and share with toy servers and third parties, and whether those third parties are ads or analytics servers.

4.1.2 Examining Security Measures

Our framework aims to evaluate what types of data are collected by the toy and its companion apps, and the security measures that have been taken to protect them. Data can be PII, in which case measures must be taken to protect its confidentiality, or it can be non-confidential content shown in the UI, like toy name and images, which must be protected from tampering.

Data is at risk of exposure when in transit and in storage. For a smart toy, transit of data, including PII, can occur between the toy and a companion app, a companion app and a remote server, or, for toys with Internet connectivity, the toy itself and a remote server. Additionally, data may be stored locally on the toy or on the device hosting the companion app. To investigate the confidentiality and integrity of data, we assess the security of communications channels between each pair of endpoints and local data storage mechanisms.

We determine the protocols used to communicate with remote hosts, and whether they are encrypted, for instance using TLS, and assess the likelihood and severity of active attacks, from replacing user interface elements in the app with inappropriate content to remotely controlling a toy.

4.1.3 Potential Vulnerabilities

We define a list of potential vulnerabilities and assess the smart toys against them. These vulnerabilities are grouped by the proximity required to exploit them, whether physical, nearby, or remote access.

Physical access. The following vulnerabilities can be exploited by an adversary with physical access to the toy.

No-local-PII-protection. An adversary with physical access to the toy can retrieve PII stored locally in the toy flash or by the companion app, leaving it unprotected in the event the toy is lost, stolen, or when sold unsanitized.

Unauthorized-config-physical. An adversary with physical access to the toy can configure it to maliciously forward PII to their account or issue harmful commands to the child.

Nearby access. These vulnerabilities require an attacker to be within close proximity to the toy, for instance within WiFi or Bluetooth range.

Unauthorized-use. This vulnerability allows an attacker to use the toy, for instance to remotely control it.

Unauthorized-config-nearby. A nearby attacker can download the companion app, connect to the toy, and maliciously configure it.

Unencrypted-hotspot. Some toys can toggle their WiFi adapter into access point mode to allow a companion app to connect directly to the toy and configure it. A toy's hotspot should authenticate devices that attempt to connect, and force transmission of data over an encrypted channel. Otherwise, an adversary who connects to a toy's open hotspot can launch a MitM attack and sniff the communication to retrieve personal information in plaintext or maliciously configure the toy.

Insecure-Bluetooth-practice. Using public, static Bluetooth MAC addresses allows a toy, and the child using it, to be tracked persistently [42]. In addition, accepting unauthorized Bluetooth connections to the toy allows adversaries to connect to the toy and change its behavior, or launch a MitM attack to sniff information transmitted between the toy and the app [4]. Bluetooth Low Energy (BLE) privacy resolves these two problems by MAC addresses randomization and whitelisting [35].

Always-on. Toys that do not have power switches and include sensitive sensors or are connected to the Internet increase their exposure to potential attack, for instance by continuously broadcasting a static MAC address.

Remote access. The following vulnerabilities expose the toy to remote attacks.

Online-password-bruteforce. Some toys allow Web login to the parent account. Login pages that allow unlimited number of password trials could allow an adversary to brute force passwords, particularly if they are weak.

No-remote-PII-protection. This vulnerability allows adversaries to access PII remotely, putting confidential data at risk.

Weak-parental-PII-control. Some toys do not allow parents to review or delete PII collected from their children, such as voice recordings or pictures.

Exposure-to-vulnerable-third-parties. Third parties with whom PII is shared, such as ads and anaytics server, may have inadequate measures to protect collected data.

Unencrypted-comm-channels. Information exchange between different parties (toy, app, and hosts) may be susceptible to interception through a MitM attack if it is not conducted over encrypted channels, such as TLS.

Insecure-session-cookies. Session cookies are used to automatically log users in to their accounts. If they are not adequately secured, for instance if they do not expire or the secure flag is not set, an adversary may gain unauthorized access to the parent account and to the child's profile.

Insecure-TLS-practices. TLS vulnerabilities [92] may result from using weak cipher suites, old TLS versions such as SSL 3.0, and vulnerable extensions. We examine servers contacted by the toy against protocol vulnerabilities such as POODLE, CRIME, Heartbleed, and Ticketbleed [80], and assess server certificates for security issues, including using short cryptographic keys and certificate mismatch.

URL-redirect. Web servers that are not hardened against a URL redirect vulnerability could allow an adversary using social engineering techniques to redirect users to phishing websites to steal their credentials. An adversary could send links to users that appear legitimate (for instance, containing the correct domain name), but that use special characters to redirect the user to another, malicious domain. Servers should prevent URL redirect, or at least whitelist accepted URLs.

4.2 Methodology and Experimental Setup

Our framework for smart toys and companion apps combines our own mechanisms with tests performed through existing test suites. Using this framework, we analyze five well-known smart toys, described in Appendix A, their companion apps, and the remote servers they communicate with against potential vulnerabilities. To obtain data for our analysis, we systematically run the toys through typical use case scenarios, using real mobile devices to circumvent evasion techniques apps may use to avoid detection of suspicious activity. Existing app analysis tools are used to detect the servers the toys communicate with, the PII they transmit, and the security measures applied by toy makers to protect personal information. We augment these wherever suitable by applying reverse engineering mechanisms, network traffic analysis, retrieval of certificate private keys, and leaking protection passphrases.

4.2.1 Hardware Setup

Our experimental setup uses a PC hosting Windows 10 professional 64-bit. The PC is configured to use a MediaTek 802.11G WiFi adapter in access point mode, using the 2.4 GHz frequency band (the band used by the toys we examine), and a second NIC with Internet connectivity. We examine the companion apps using a Samsung N7100 hosting Android 4.4.2. This version of Android is ranked among the top four Android versions used in the wild [11]. As an older version of Android, it is illustrative of how toys deal with older cipher suites, TLS versions and TLS protocol vulnerabilities.

4.2.2 Network Analysis

Our experimental setup uses Wireshark [9] to sniff network traffic initiated by the toys and their companion apps. We configure a Man-in-the-Middle (MitM) proxy to intercept TLS traffic to and from the companion app and, wherever applicable and/or possible, the toy itself. This entails installing a certificate to the mobile device's trusted store corresponding to the MitM proxy used, ensuring that the smartphone accepts certificates signed by the corresponding private key. In lieu of establishing a TLS connection between the device and the remote server, two TLS connections are formed instead: one between the end device and the proxy using the installed certificate, and another between the proxy and the remote server. In this configuration, the proxy acts a server to the end device, and a client to the remote server. To decrypt traffic, we use Burp Suite [5] as a MitM proxy and adds the Burp Suite CA certificate to the smartphone CA store. We address special cases where this approach was insufficient to intercept and decrypt TLS communications. All such interventions were performed on Android-based client apps that were reverse-engineered. The Qualys SSL Labs [6] testing suite was used to perform TLS analysis on the server side, and we examined packet captures in Wireshark to analyze TLS practices on the client app.

4.2.3 Ads and Analytics Analysis

Wherever possible, we intercept data between the companion app and third party ads and analytics servers (Internet-capable toys we examined do not communicate with ads and analytics services). We investigate the analytics servers to detect any instances of unique identifiers correlation with the users' personal information such as email address. We also inspect usage of cross-platform cookies such as DSID and IDE [7] cookies which are used to distinguish users across different platforms.

4.2.4 App Analysis

All companion apps were analyzed manually (as opposed to using, for example, the Monkey automation tool [8]). We mimicked common use case scenarios with the goal of triggering app UI events looking for signs of PII leakage, weak security measures, or potential vulnerabilities, and captured data from the interaction with the toy and its companion app.

Modify custom CA store. When the companion app uses a custom CA store to verify server certificates, we apply the following methodology to force the app to accept the MitM certificate: (a) Decompile the app using Apktool [2] to retrieve the CA store from the app's assets directory, (b) Patch parts of the app smali files to force revealing the CA store password, (c) Use the password to access and update the custom CA store, (d) Use the keystore explorer tool to add the MitM certificate, and (e) Replace the CA store in the assets directory of the app with the new store, rebuild the app using Apktool, re-sign and verify the patched apk file, and use the adb tool to reinstall it. As a result, the app accepts the MitM certificate.

Exfiltrate client SSL certificate. In cases where the companion app authenticates to the server using a client Bouncy Castle file (PKCS#12), which encompasses the client's public key and the client's private key, we reverse engineer the app to exfiltrate the PKCS#12 file and the passphrase used to protect it. We then add the PKCS#12 file (the certificate and the private key) to the interception proxy. This allows the interception proxy to authenticate to the server.

Bypass certificate pinning. In cases where the companion app uses certificate pinning to refuse server certificates signed by any CA other than the one with the pinned certificate in the app, we patch the corresponding parts of the app smali files to disable this feature and intercept the communication.

4.2.5 TLS Vulnerabilities Analysis

We assess toys, apps and server SSL practice to determine potential TLS protocol vulnerabilities, and use Qualys SSL Labs to determine TLS server parameters, including supported TLS versions, cipher suites and extensions.

4.2.6 Bluetooth Analysis

BLEscanner [3] is used to examine the Bluetooth connection between the toy and the companion app, and determine whether the toy's Bluetooth MAC address is persistent or dynamically changing. We investigate how an adversary could tamper with the communication or gain unauthorized access to Bluetooth parameters to conduct MitM or other attacks [51].

4.3 Results

All toys we examined collected PII, and most of them communicated with one or more ad and analytics servers. Alarmingly, the toys geared towards younger children collected PII more aggressively and in general secured it more poorly, while toys geared for older children collected less PII, generally did not require account creation, and had more thoughtful security measures in place to protect them. Toys geared to younger children collected more intrusive PII, such as cross-service and cross-platform unique identifiers and user and devices fingerprinting information, and collected it more frequently.

Table 7: Types of persistent user identifiers collected by smart toys

	Multi-session tracking	Multi-service tracking	Multi-platform tracking
Toymail	✓	✓	✓
Wiggy Piggy Bank	✓	✓	
Hello Barbie	✓	✓	
Wowwee Chip	✓	✓	
Cloudpets	✓	✓	✓

Table 7 shows the persistent user identifiers collected by the smart toys and their companion apps. Some identifiers captured in the transmissions were obfuscated, or their purpose was not clear. In these cases, we have made conservative assumptions about their purpose; if they appear across multiple sessions, like an app ID, then they are deemed to track the user across multiple sessions of the same app. All the Android apps we tested transmit the Google advertising identifier, even when ad tracking is disabled in the smartphone. These apps, and those that transmit a hardware identifier, are deemed to track users across multiple services on the same device. Finally, apps that use cookies such as DSID and IDE on Android [7], or transmit email addresses or other identifying information about the user to analytics servers, are deemed to track users across multiple platforms.

Table 8 shows the ads and analytics servers contacted by toys and companion apps. Four out of five toys we examined send data to two or more analytics servers. The set of toys we examined are vulnerable to some form of attack, whether through physical or nearby access (within WiFi or Bluetooth coverage), or remotely, such as over HTTP, as shown in Table 9. In the following subsections, we discuss our findings.

Table 8: Ads and analytics services contacted by smart toys and companion apps

Smart Toy	tangible-analytics.appspot.com	googleads.g.doubleclick.net	googlesyndication.com	googleadservices.com	google-analytics.com	data.flurry.com	e.crashlytics.com	unity3d.com	hockeyapp.net	api.branch.io	api.segment.io	wzrkt.com	ads.mopub.com
Hello Barbie													
Toymail							✓			✓	✓	✓	
Wiggy Piggy Bank					✓			✓					
Wowwee Chip						✓	✓						
Cloudpets		✓	✓	✓									✓

4.3.1 Toymail

PII collected and transmitted. The companion app sends PII to app.toymailco.com including login credentials, email address, parent profile ID, child name, child date of birth, child photo, friends profiles IDs and names, friends profile pictures, and voice messages. The app communicates with multiple ads and analytics services which collect personal information as follows. E.crashlytics.com collects smartphone device information and app crash information, along with smartphone hardware ID, Google ads ID, and app installation ID. Api.branch.io collects smartphone device and OS information, Toymail user ID, smartphone hardware ID, smartphone fingerprint ID, IP address, identity ID, and email address. Api.branch.co verifies whether the smartphone is a real device or an emulator. Wzrkt.com sets a unique ID for the user which it collects while the app is running, allowing tracking the user across different sessions. It also collects the Google Ad ID (allowing tracking the user across different services), smartphone device information, and telephone

Table 9: Attacks by proximity

	Physical Access			Nearby				Remote Access							
Smart Toy	No-local-PII-protection	Unauthorized-config-physical	Unauthorized-use	Unauthorized-config-nearby	Unencrypted-hotspot	Insecure-Bluetooth-practice	Always-on	Online-password-bruteforce	No-remote-PII-protection	Weak-parental-PII-control	Exposure-to-vulnerable-third-parties	Unencrypted-comm-channels	Insecure-session-cookies	Insecure-TLS-practices	URL-redirect
Toymail	✓	✓					✓			✓	✓	✓	✓	✓	✓
Wiggy Piggy Bank		✓	✓	✓		✓		✓					✓		✓
Hello Barbie		✓		✓	✓			✓							✓
Wowwee Chip	✓		✓			✓									
Cloudpets		✓				✓		✓			✓	✓	✓	✓	✓

service carrier information. Api.segment.io collects Toymail user ID, name, email address, device hardware ID, Google ads ID, smartphone device and OS information, data network device status (WiFi, Bluetooth, cellular), and phone carrier name. None of these ads and analytics servers respects the device-wide ad tracking setting, and all collect PII regardless of whether it was disabled. Unsurprisingly, we found that using the toy resulted in targeted ads even on different platforms by setting unique identifiers (e.g., identity ID) and correlating them with other personal information (e.g., email address), as shown in Figure 3 which depicts a Toymail ad in a PC hosting Windows, on a different WiFi network.

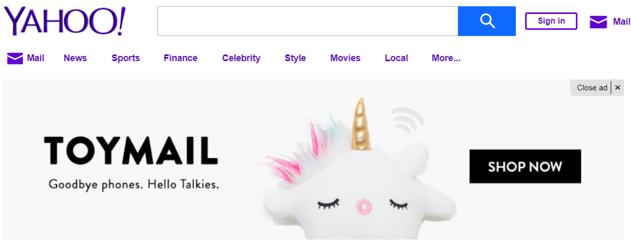


Figure 3: Tracking example: Toymail tracking users across different platforms - Toymail ad shown in a PC while the user is browsing Yahoo!

Once the toy connects to the WiFi network, it communicates with imp07a.boxen.electricimp.com to receive a list of allowed contacts. When the child selects a contact and presses the message button, the toy begins recording. When the child presses the message button again, the toy transmits the voice recording to the server, which stores them in Amazon AWS storage. The toy keeps up to ten voice messages locally which are received from parents and contacts. It also stores Toymail child ID, toy ID, contacts' names, and contacts' Toymail IDs, in addition to WiFi SSID and password.

Security measures. The app connects to all servers over TLS. The app.toymailco.com server certificate is signed by Go Daddy and uses a 2048-bit RSA key. The app maintains its own list of strong cipher suites, and uses certificate pinning to prevent MitM attacks. It stores hashes of the certificate chain public keys to bind the server certificate with the original issuer. However, the toy suffers from a number of *Insecure-TLS-practices*. The imp07a.boxen.electricimp.com server certificate is signed by the custom root CA impca, and both certificates are sent to the toy. The server certificate is valid for 20 years and it uses a 1024-bit RSA key whose certificate common name is different from the server's domain name. The toy uses a client certificate to authenticate to the server. It is signed by impca, uses a 2048-bit RSA key, and is valid for 100 years. The app communicates with two analytics and ads servers vulnerable to RC4 and POODLE attacks, with support for SSL 3.0 and weak cipher suites, exposing Toymail to *Exposure-to-vulnerable-third-parties*. Neither the session cookie expiration date nor secure flag are set, leaving the toy vulnerable to *Insecure-session-cookies*.

Toymail is rated to offer *Weak-parental-PII-control*. When a child sends recordings to contacts other than the parent, they are stored in the contact's account, leaving the parent without means to review or delete them. Even when a parent blocks a contact from the child's contact list, that contact can still access the child's personal information. The server sends the child's personal information to the contact's app accompanied by a "blocked" flag (equal to "-1") indicating that the recipient has been blocked. The app removes the entry corresponding to the child from the app interface so they can no longer exchange voice messages; however, the contact still receives the child's personal information each time the app is launched including: name, date of birth, photo, voice recording of the name, parent Toymail ID, toy ID, wake up time, sleep time, and "isOnline" flag showing the state of the toy.

With physical access to the toy, anyone can access these messages or even add contacts to the contact list without authentication, making it vulnerable to *No-local-PII-protection* and *Unauthorized-config-physical*, respectively. In fact, there is no mechanism to delete locally stored voice recordings other than configuring the toy to use another account.

Toymail is designed to be *Always-on*. Switching off requires removing the batteries, accessible only by removing the back cover using a screwdriver. Given that the toy includes sensitive devices like a microphone and is connected to the Internet, we deem this a significant potential threat to children's privacy and security.

4.3.2 Wiggy Piggy Bank

PII collected and transmitted. The app communicates with the server api.kii.com, sending parent login credentials (email address and password or session cookie), user ID, children's profile pictures, tasks set by parent, tasks completed by children, and children's goals. The app also communicates with ads and analytics services. Unity3d.com collects an app ID, device unique ID, user ID, session ID, and device and OS information. Moreover, even when ad tracking is disabled, unity3d.com still collects an advertising ID uniquely identifying the user across different services, and app usage information including app start time and duration. The app also connects to google-analytics.com, which collects analytics data and applies obfuscation techniques before sending them.

Security measures. The app's connection to api.kii.com appears to use TLS in a secure way. The app relies on the OS trusted store to check the server certificate. The server uses a certificate signed by well-known certificate authority (Go Daddy Secure Certificate Authority - G2) and signed by a 2048-bit RSA key using SHA256. The server public key uses a 2048-bit RSA key and the certificate is valid for a limited period (expiring in October 2019). The server has six cipher suites that support forward secrecy and they are at the top of the server's cipher suites list. The server is patched against vulnerabilities including POODLE and Heartbleed, Ticketbleed, and does not support TLS compression, which causes vulnerabilities like CRIME. Although the app uses HTTPS to communicate with the server api.kii.com, the server does not force using HTTPS in the case of flipping HTTPS to HTTP, leaving all communication unencrypted, and making the toy vulnerable to *Unencrypted-comm-channels*. We use the Burp Suite repeater tool to flip all app requests to the server on the fly to HTTP instead of HTTPS, and as a result, the app connects to the HTTP version of the server and receives PII in plaintext.

A determined adversary can take over a parent's account in several ways. The first scenario is a result of *Insecure-session-cookies*. All session cookies expire January 19, 2038 - a long period of time during which any adversary who steals the cookies may use them to access the parental account. In a second scenario, an *Online-password-bruteforce* vulnerability allows an attacker to gain access to the parent's account. This was verified by using the Burp Suite intruder tool to brute force accounts owned by us. However, the most serious vulnerability we uncovered, *Unauthorized-config-nearby*, enables an adversary to access PII or assign tasks to a child simply by being within Bluetooth coverage. Anyone within Bluetooth range can install the companion app, pair with the toy, and issue tasks, both new and predefined. These new tasks would appear on the child's account just as if they had been assigned by the child's parents. The toy announces predefined tasks in what to the child is a familiar and trusted voice, increasing the likelihood that the child will fall prey to the attack. The attacker may also add a valuable (but fake) reward to encourage the child to perform the task. An adversary needs to take over the parent's account (for instance by brute forcing the parent's password) to be able to assign new tasks to the child, but does not need to do so to assign predefined tasks, including some that encourage the child to leave the house like "pull the weeds", "rake the leaves", "take out the trash", and "walk the dog". The toy also broadcasts a static Bluetooth MAC address, making it vulnerable to *Insecure-Bluetooth-practice*.

4.3.3 Hello Barbie

PII collected and transmitted. The companion app transmits the following information to api.2.toytalk.com: the smartphone manufacturer, device name and install ID (unique per device), account ID (unique per user), app version, user consent flag, OS name, and OS version (see also Table 8). The app also sends login information including email address and password, as well as the child's date of birth and any other important dates the parent has specified. We did not observe traffic to any ads or analytics services. The toy does not store the child's voice recordings, but streams them directly to the cloud. The toy stores the WiFi network SSID and passphrase, as well as the parent account ID that is used to access the parent profile and listen to recordings. On the other hand, the app stores sensitive information in plaintext, including a session cookie authenticating the user to the toy server, the parent email address, and profile ID.

Security measures. The app does not rely on the OS CA store to verify the server certificate. The custom CA store, which is included in the app's assets directory, contains one certificate for "Toytalk ca", a self-signed certificate used to sign the server certificate. In addition, the app uses a Bouncy Castle PKCS#12 certificate to authenticate to the server. The app connects only to servers in its whitelist; at the time of testing, these were test.2.toytalk.com and api.2.toytalk.com, although we did not observe any communication with the first server, which seems to be for testing purposes. The app communicates with api.2.toytalk.com over TLS using the OS TLS implementation, which may contain weak cipher suites or vulnerable practices specifically in older OSes such as Android 4.4.

There are several scenarios in which an adversary can exploit *Insecure-TLS-practices* to steal the parent's credentials and access the child's recordings. The client, server, and self-signed root certificates use RSA with 1024-bit keys which are being phased out [10]. The server certificate private key can be used to decrypt the negotiated session key, and the root CA key can be used to sign certificates as api.2.toytalk.com to steal the parent's credentials.

The toy communicates with three servers: firmware.toytalk.com, puppeteer.toytalk.com, and storage.toytalk.com. When Hello Barbie is switched on, it connects to the WiFi network and initiates a connection to firmware.toytalk.com over TLS. While the child presses and holds the "talk" button, Hello Barbie streams the child's speech over a TLS-protected channel to puppeteer.toytalk.com. Once the "talk" button is released, puppeteer.toytalk.com sends the link to the new recording to storage.toytalk.com, where it is stored. Analysis of the three servers shows that two of them, namely firmware.toytalk.com and storage.toytalk.com, support weak cipher suites (TLS_RSA_WITH_3DES_EDE_CBC_SHA with TLS 1.0). Bhargavan et al. [18] found that ciphers using 64-bit blocks (e.g., 3DES) are vulnerable to secret key disclosure.

The official website toytalk.com can be used by parents to review the child's recordings. The server certificate uses an RSA 2048-bit key and is signed by Amazon using SHA256 and an RSA 2048-bit key. The server does not support weak cipher suites and uses cryptographic libraries patched against known cryptographic attacks. However, there is no restriction on the number of login attempts, and the login can be bruteforced. We exploited this *Online-password-bruteforce* vulnerability during the course of testing to bruteforce our own account. This flaw was originally reported in January 2016 [82] and our experiments show that it is still not fixed.

To investigate communication between the toy and the app, we use the following specific setup for Hello Barbie. We use two WiFi adapters on our test machine, one in AP mode as an impostor Hello Barbie, and the second to connect to the real Hello Barbie, exploiting its unprotected hotspot. We route all traffic from the impostor to the real toy. In the app, both toys, real and impostor, are listed, and we connect to the impostor. The smartphone hosting the app, meanwhile, is configured to forward all traffic through a Burp Suite proxy. This configuration requires the app to be patched to accept certificates signed by Burp Suite CA, and the app client certificate bundle added to Burp Suite as the client certificate. This allows the app to authenticate itself to the toy, enabling us to intercept communication between them. The toy authenticates to the app using a certificate issued to 192.168.10.1, signed by “ToyTalk CA”, and valid to 2030. Once the TLS handshake is complete, the app sends the parent account ID and WiFi configuration, including WiFi SSID and password, to the toy.

The toy can be configured using any companion app, and no parent authentication is required to pair with the toy. Because the toy uses the parent’s account ID as the sole way to relate the child’s profile, including voice recordings, to the parent, an adversary need only reconfigure the toy to use the adversary’s account to access all subsequent recordings made by the child.

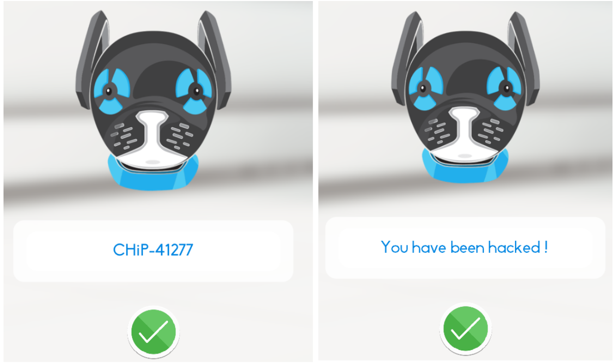
There are several scenarios in which the toy is vulnerable to *Unauthorized-config-nearby*. In the first scenario, an adversary who knows the WiFi credentials can configure the toy with their account ID; in the others, the adversary does not require the WiFi credentials. While the toy is in pairing mode, it broadcasts an open network with SSID "Barbie-950", making it temporarily vulnerable to *Unencrypted-hotspot*. An unencrypted hotspot could allow an adversary to conduct ARP spoofing, assuming a MitM position between the app and the toy. In this scenario, the legitimate toy and the app will unwittingly treat the adversary as the destination for all traffic, allowing the adversary to sniff the parent account ID and the WiFi credentials. We deem this a minor risk as the vulnerability is only present for a limited duration (during pairing), and an attacker must be within close proximity. However, the toy can be accidentally put in pairing mode during the course of normal play, as it requires the child to simultaneously press two easily accessible buttons on the toy. Once it is in pairing mode, an adversary within wireless range can download the app, configure the toy to connect to a different WiFi network under their control, and then configure it to use the adversary's account.

4.3.4 Wowwee Chip

PII collected and transmitted. The Wowwee Chip companion app sends analytics and bug-related data to Flurry Analytics and Crashlytics. Flurry Analytics collects a hardware unique ID and ad ID, as well as toy usage information, smartphone and OS information, telephone carrier information, approximate location, and smartphone state information including battery remaining percentage, battery charging state, memory available, external and internal disk sizes and available spaces. Crashlytics collects ad ID, app installation ID, and smartphone device information, in addition to other obfuscated data.

Security measures. Wowwee Chip adopts strong TLS practices. The app does not use TLS 1.0, and the servers are patched against POODLE, Downgrade, and Heartbleed attacks. Wowwee Chip is not vulnerable to CRIME since data compression is not supported in the app or the servers. The app uses its own list of cipher suites which does not contain weak ciphers or short keys, and it supports only TLS version 1.2. All cipher suites support forward secrecy. Flurry Analytics and Crashlytics certificates use RSA 2048-bit, and are issued by well-known issuers with secure signature algorithms. Certificates validity periods are limited, and they are not revoked. Although both servers support TLS 1.0 and a weak cipher suite (TLS_RSA_WITH_3DES_EDE_CBC_SHA112), since they are not supported by the app, we consider the app not to suffer from *Insecure-TLS-practices*.

On the other hand, Wowwee Chip is vulnerable to *No-local-PII-protection*. All components of the toy can be paired with the RamBLE utility without authentication, allowing us to access and modify toy information that includes manufacturer name, model number, serial number, firmware revision and battery level. An adversary could also modify the name of the toy to something inappropriate, which would appear in the app when the child or parent uses it to play with the toy. Figure 4 shows the Wowwee Chip app displaying a maliciously altered toy name. The toy and all its companion components (Smart Bed, Smart Ball, Smart Band) use static Bluetooth MAC addresses that allow child tracking, leaving it vulnerable to *Insecure-Bluetooth-practice*. It is likewise vulnerable to *Unauthorized-use* since the connection between the companion app and the toy is over unencrypted BLE.



CHiP-41277

You have been hacked !

Figure 4: Maliciously changing Wowwee Chip's name

Any installed instance of the app can pair with the toy and have access to full toy functionality. A child can play with Wowwee Chip directly without the need for an app, leaving the toy available for Bluetooth pairing. This could be exploited maliciously if an adversary is within Bluetooth range (about 33 metres). Such an adversary could control the toy, navigating it and making it perform activities such as Yoga or Wanna Play that could injure a child, especially taking into account the weight of the toy at 2.7 kg. An adversary could also make the toy bark continuously by adjusting the wake up alarm, possibly scaring the child.

4.3.5 Cloudpets

PII collected and transmitted. The Cloudpets companion app connects to the server parse-cloudpets.spiraltoys.com and transmits parent and child names, pictures, and dates of birth; parent email address; friend names and profile photos; and child voice messages, which are stored in the parent account in the cloud. It connects to ads and analytics servers which collect PII as follows. Googleads.g.doubleclick.net sets DSID and IDE cookies, which link user activity and feature targeted ads across different platforms [7].

Ads.mopub.com collects the unique device identifier and sends it after hashing using SHA-1. The app also connects to googleadservices.com and googlesyndication.com, which initiate requests to googleads.g.doubleclick.net. The Cloudpets companion app sends these unique identifiers even if the user disables ad tracking.

Security measures. The toy server suffers from *Insecure-TLS-practices*. It supports only TLS 1.0, and several anonymous cipher suites (e.g. TLS_ECDH_anon_WITH_RC4_128_-SHA) which allow accessing the server without authentication and expose the app-server connection to the risk of MitM attacks. Moreover, the server is vulnerable to RC4, and uses weak key exchange and common DH primes, increasing the risk of losing forward secrecy for transmitted PII.

PII stored remotely, including audio files and photos, can't be accessed without authentication. All PII is transmitted over encrypted channel; however, the server does not force using HTTPS. Flipping HTTPS to HTTP sends PII in plaintext, exposing the toy to the *Unencrypted-comm-channels* vulnerability. The toy suffers from *Insecure-Bluetooth-practice* as it uses a static MAC address and it does not apply BLE privacy; a MitM could access communication between the app and the toy.

The app displays ads that cover the entire screen at app launch and periodically during use; some of them force users to wait for several seconds before allowing them to close the ads. Moreover, the app features a persistent ad bar at the bottom or top of the interface which continuously displays ads to the user. While many free standalone smartphone apps are monetized through ads, it is unusual to observe aggressive (or any) ad display in the companion app of a purchased toy, and Cloudpets is unique in this regard among the toys we examine.

4.4 Conclusion

In this chapter, we examine the security and privacy practices of five popular smart toys on the market. Our findings show widespread use of data collection by toy makers and third party analytics servers, with almost all toys embedding analytics services within their apps, and many of them transmitting advertising identifiers even when ad tracking is disabled by the user. Many toys do not take adequate measures to protect sensitive data, and some toys do not protect locally stored PII at all, making it trivially accessible to anyone with physical access. Children's physical privacy may also be compromised, as all Bluetooth-capable toys we examine advertise a static MAC address, making it possible to track them through physical space. Finally, several toys are vulnerable to exploits that can result in handing an attacker control of the toy or some of its functionality, potentially causing harm to a child.

Chapter 5

Conclusion and Future Work

To conclude, we presented two frameworks to analyze security and privacy practices of smart toys. The first framework is analytical and we used it to evaluate privacy practices of a representative set of 11 smart toys and their companion apps by analyzing their privacy policies and terms of use documentation. We highlighted several concerns in the privacy practices of these smart toys regarding PII collection, third-party data sharing, web tracking, and data storage location. Moreover, static analysis for the companion apps reveals that some toys' companion apps are over-privileged, others collect unnecessary personal information, and some of them get involved in suspicious activities such as abusing the telephony service. In our second framework, we investigated the security and privacy practices of five popular smart toys using our experimental setup. Our analysis shows excessive data collection by toys and third parity analytics servers and many of the toys transmit users unique identifiers even if ad tracking is disabled by the user. We also noticed that many toys do not use proper security measures to protect the collected private information. Several toys are vulnerable to exploits that can allow adversaries to take over control of the toy in a way that can cause potential harm for the children.

5.1 Recommendations and Best Practices

Just as manufacturers of traditional toys must follow mandated safety standards, smart toy makers should additionally adhere to security and privacy guidelines [63]. Based on the experience gained throughout this research, in here, we list some recommendations for makers of connected toys.

5.1.1 Security-Related Practices

Toys and companion apps must use a custom list of strong (or at least not weak) cipher suites and recent TLS versions (e.g., TLS 1.2) to avoid attacks including Beast and POODLE. Servers also are recommended to be patched against known TLS protocol vulnerabilities including POODLE, Downgrade, RC4, Heartbleed, and Ticketbleed vulnerabilities. Wherever possible, host whitelisting should be used, hard coded in the smart toy firmware and the companion app, as it can mitigate against phishing attacks. Certificate pinning can also help prevent MitM attacks. Toys with wireless capabilities should take particular care to secure pairing and connection. If the toy uses Bluetooth, it should use encryption. Using open Bluetooth can allow unauthorized access to the toy and PII stored on it. Toys should also use dynamic Bluetooth MAC addresses to avoid the possibility of tracking children's locations. WiFi-enabled toys should be provisioned in a secure manner, with WiFi credentials supplied securely over an out-of-band channel. Moreover, toys manufacturers should properly mention security measures that have been taken to protect the collected information in the legally-binding documentation.

5.1.2 Privacy-Related Practices

Toys manufacturers should provide complete information about data storage location and legal compliance. They should also restrict information collection to those that are needed for the purpose of the toy functionality (unnecessary collection of privacy-sensitive information can pose questions about the intention of collecting these sensitive information). In all cases, smart toys companies must document personal information they collect in the privacy policy or terms of use documentation. Toys should not store personal information (e.g., voice recordings, photos, videos, personal information that identify users/children) internally on the toy's flash memory; if it is necessary for toy functionality, then they should be stored encrypted. Transmission of personal information should always occur over secure communication channels.

Toy manufacturers that provide links to PII, such as voice recordings, should also grant parents the ability to select whether the PII is public or private. Even when links are randomly generated, if they are sufficient to access PII then we deem it a threat to privacy. We also recommend allowing a default setting the parent can configure to determine whether such links are public or private. Moreover, parent control should be provided to allow parents to review or delete collected personal information by the toy and its companion app. Toys manufactures should also provide parents with a dedicated privacy support via an email address or a specific web page.

In addition, companion apps should not be over-privileged (i.e., more permissions declared in the Manifest than the app needed or used). Over-privileged apps can pose privacy risks, as they may request extraneous permissions, which can be used to access sensitive information. We strongly caution against ad display in companion apps, especially as these apps form part of a purchased toy and thus should not require support through ad revenue. In addition, ads and analytics services should be used sparingly with children [15], and when used, should respect users' request to disable ad tracking. Specifically, we observed multiple instances of Android apps flouting the user's preferred ad tracking setting, and sending the ad identifier regardless of whether it was disabled. In the Android model, a single system call returns both the ad identifier and the limit ad tracking setting, and the developer is assumed to not abuse them [1].

5.2 Future work

In what follows, we provide a summary of some possible future work directions:

- Firmware analysis: Firmware analysis may show whether toys store personal information locally on the toy and if the information is encrypted or in plaintext, an important consideration especially for used or lost toys. Firmware analysis may also help determine whether the firmware can be modified maliciously, such as to establish rogue connections with an adversary.

- Our analytical framework for analyzing privacy polices and terms of use documentation need to be automated for better scalability (as opposed to our current manual static analysis).

- It is useful to create an experimental setup that can examine companion iOS apps and provide a comparison between security and privacy practices in case of using Android apps versus iOS apps.

Appendix A

Selected Smart Toys

We investigate a wide spectrum of toys available on the market, varying by target age range and functionality. The toys we selected span multiple target ages, from Hello Barbie and Smart Toy Monkey, geared towards younger audiences, to what are generally thought of as STEM (Science, Technology, Engineering, and Math) toys aimed at children 8+, such as Sphero BB-8, and Anki Cozmo. Toy functionality encompasses AI capability, such as voice and image recognition; sensors, like mic and camera; mobility; and wireless communication, like WiFi and Bluetooth. These toys are described below.

Toymail allows children to exchange voice messages with parents, relatives and friends using the companion app or another Toymail. Parents can configure the toy with an approved list of contacts. Voice alerts notify children when they have new messages, and they can click the play button to listen.

Wiggy Piggy Bank is a Bluetooth-connected toy that allows parents to set tasks, create goals, and send rewards to their children through a companion app. Children are notified they have new tasks either by the toy, which gives alerts and states the tasks, or by checking their accounts through the app. Children confirm through the app that they have completed a task, and upon parental approval virtual funds are transfered to their account. A child can set goals (e.g., buying a mobile phone for $100). When the balance of the child's virtual account achieves the target value of the goal, the child may redeem it from the parent.

Hello Barbie is a smart doll designed to conduct interactive conversations with a child. The toy connects to the Internet directly to send the child's voice recordings to the toy server, which applies voice recognition techniques to respond to the child. The toy has a companion app that a parent can use to configure the toy.

Sphero BB-8 is a Star Wars character-branded spherical robot that can be remotely piloted via a companion app. The Sphero BB-8 is one of a line of spherical robots designed for both recreational and educational use. It can be controlled programmatically using the beginner-friendly Sphero SPRK Lightning Lab smartphone app, or a variety of both official and unofficial SDKs for multiple platforms. The toy features an on-board gyroscope and accelerometer, and is designed to be controlled by compatible BLE-enabled devices. Although Sphero BB-8 has been designed to be controlled in a variety of ways, including programmatically, our study focused primarily on the most common (and, to a child, intuitive) scenario: running the companion Sphero BB-8 app on a smartphone.

Anki Cozmo is an intelligent programmable bot with onboard speaker, camera, and inbuilt facial recognition.Anki Cozmo sings, plays games and observes its surroundings. The toy comes with 3 programmable, LED-colored cubes. It is interactive, and can be played with using a companion app, or programmed via a Python-based SDK, available on

GitHub.

Smart Toy Monkey is one of a related line of Smart Toys by Fisher Price. The toy is interactive, featuring a microphone, speaker, and voice recognition, and an accelerometer that detects if it is being thrown in the air. The toy has limited image recognition, mainly for use with accompanying activity cards that can be used in lieu of the companion app.

Wowwee Chip is a robot dog that responds to voice commands and touch. Wowwee Chip comes with a Smart Ball and Smart Bed that doubles as a charging dock, and responds to commands from a companion watch called a Smart Band or the Wowwee Chipmobile app.

Cloudpets allows exchanging voice messages between children, parents, and friends. The toy does not connect to the Internet directly, but connects to a companion app through Bluetooth. A parent uses the app to create parent and child accounts, and links the toy to the child's account. The toy receives a link request, appearing as red heart pulses; pressing a button on the toy's hand accepts the connection. Through the parent account, parents can accept or reject messages sent to or from the toy.

CogniToys Dino: Powered by IBM Watson and Friendgine technology, CogniToys Dino can interact with children by telling stories, make them laugh by cracking jokes and playing games.

Edwin the Duck: A toy duck that tells interactive stories, sings and plays stimulating games via the companion app.

My Friend Cayla: A doll that utilizes speech recognition technology and Internet connectivity to answer questions on various topics.

I-Que Robot: An intelligent robot that responds to queries, dances on music and allows children to have real-time conversations with the toy.

Zenbo: A smart robot that can roam around the house taking photos, making video calls and playing songs. It can tell entertaining stories to children.

Appendix B

Smart Toys Privacy Policy Links

Table 10 shows links to the toys' privacy policies and terms of use documentation.

Table 10: Links to toys' privacy policies and terms of use

Product	Policy link	ToU link
Hello Barbie	https://www.toytalk.com/hellobarbie/privacy	https://www.toytalk.com/hellobarbie/terms
Toymail	https://toymail.co/pages/privacy	https://toymail.co/pages/terms
Sphero BB-8	http://www.sphero.com/privacy	http://www.sphero.com/terms
Wowwee Chip	http://wowwee.com/information/privacy	http://wowwee.com/information/warranty
Smart Toy Monkey	http://www.smarttoy.com/privacy	http://www.smarttoy.com/terms
CogniToys Dino	https://cognitoys.com/pages/privacy	https://cognitoys.com/pages/terms
Edwin the Duck	http://www.edwintheduck.com/privacy-policy	http://www.edwintheduck.com/terms-and-conditions
Anki Cozmo	https://www.anki.com/en-ca/company/privacy	https://www.anki.com/en-ca/company/terms-and-conditions
My Friend Cayla	https://www.myfriendcayla.com/privacy-policy	http://myfriendcayla.co.uk/terms
I-Que Robot	http://ique-robot.co.uk/privacy	http://ique-robot.co.uk/terms-conditions
Zenbo	https://www.asus.com/Terms_of_Use_Notice_Privacy_Policy/Privacy_Policy	https://www.asus.com/Terms_of_Use_Notice_Privacy_Policy/Privacy_Policy

Bibliography

[1] AdvertisingIdClient – Google APIs for Android. https://developers.google.com/android/reference/com/google/android/gms/ads/identifier/AdvertisingIdClient.

[2] Apktool. https://github.com/iBotPeaches/Apktool.

[3] BLE Scanner. https://play.google.com/store/apps/details?id=com.macdom.ble.blescanner.

[4] Bluetooth Low Energy Security. https://www.bluetooth.com/~/media/files/specification/bluetooth-low-energy-security.ashx.

[5] Burp Suite Scanner. https://portswigger.net/burp.

[6] Qualys SSL Labs. https://www.ssllabs.com.

[7] Types of cookies used by Google. https://www.google.com/policies/technologies/types/.

[8] UI/Application Exerciser Monkey. https://developer.android.com/studio/test/monkey.html.

[9] Wireshark. https://www.wireshark.org.

[10] Mozilla security blog: Phasing out certificates with 1024-bit rsa keys. https://blog.mozilla.org/security/2014/09/08/phasing-out-certificates-with-1024-bit-rsa-keys, September 2014.

[11] Dashboards. https://developer.android.com/about/dashboards/index.html, October 2017.

[12] Manar Alohaly and Hassan Takabi. Better privacy indicators: a new approach to quantification of privacy policies. In *Symposium on Usable Privacy and Security (SOUPS'16)*, Denver, CO, USA, 2016.

[13] Julia Angwin and Tom McGinty. Sites feed personal details to new tracking industry. *Wall Stree Journal, available at http://online. wsj. com*, 2010.

[14] Kathy Wain Yee Au, Yi Fan Zhou, Zhen Huang, and David Lie. Pscout: analyzing the android permission specification. In *ACM conference on Computer and Communications Security (CCS'12)*, Raleigh, NC, USA, 2012.

[15] M. Jill Austin and Mary Lynn Reed. Targeting children online: Internet advertising ethics issues. *Journal of Consumer Marketing*, 16(6):590 – 602, 1999.

[16] BBC. Call for privacy probes over Cayla doll and i-Que toys. http://www.bbc.com/news/technology-38222472, December 2016.

[17] BBC UK. German parents told to destroy Cayla dolls over hacking fears. http://www.bbc.com/news/world-europe-39002142, February 2017.

[18] Karthikeyan Bhargavan and Gaëtan Leurent. On the practical (in-) security of 64-bit block ciphers: Collision attacks on HTTP over TLS and OpenVPN. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

[19] Justin Brookman, Phoebe Rouge, Aaron Alva, and Christina Yeung. Cross-device tracking: measurement and disclosures. *Privacy Enhancing Technologies*, 2017.

[20] Federal Trade Commission. Mobile apps for kids: current privacy disclosures are disappointing. https://www.ftc.gov/sites/default/files/documents/reports/mobile-apps-kids-current-privacy-disclosures-are-disappointing/120216mobile_apps_kids.pdf, February 2012.

[21] Federal Trade Commission. Mobile apps for kids: disclosures still not making the grade. https://www.ftc.gov/sites/default/files/documents/reports/mobile-apps-kids-disclosures-still-not-making-grade/121210mobilekidsappreport.pdf, December 2012.

[22] Federal Trade Commission. Children's online privacy protection rule ("COPPA"), 2015.

[23] Federal Trade Commission. Kids' apps disclosures revisited. https://www.ftc.gov/news-events/blogs/business-blog/2015/09/kids-apps-disclosures-revisited, September 2015.

[24] Lara Spiteri Cornish. ‘Mum, can I play on the Internet?’ Parents’ understanding, perception and responses to online advertising designed for children. *International Journal of Advertising: The Quarterly Review of Marketing Communications*, 33(3):437 – 473, 2014.

[25] Elisa Costante, Jerry den Hartog, and Milan Petković. What websites know about you. In *Data Privacy Management and Autonomous Spontaneous Security*. 2013.

[26] Elisa Costante, Yuanhao Sun, Milan Petković, and Jerry den Hartog. A machine learning solution to assess privacy policy completeness (short paper). In *ACM Workshop on Privacy in the Electronic Society (WPES’12)*, Raleigh, NC, USA, 2012.

[27] Lorrie Faith Cranor, Kelly Idouchi, Pedro Giovanni Leon, Manya Sleeper, and Blase Ur. Are they actually any different? comparing thousands of financial institutions’ privacy practices. In *Workshop on the Economics of Information Security*, 2013.

[28] CTVNews. Children becoming targets of identity theft, 2014.

[29] 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 *The ACM international conference on Ubiquitous computing*, pages 105–114, 2009.

[30] Danielle L Dobbins. *Analysis of security concerns and privacy risks of children’s smart toys*. PhD thesis, Washington University St. Louis, St. Louis, MO, USA, 2015.

[31] Steven Englehardt and Arvind Narayanan. Online tracking: A 1-million-site measurement and analysis. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1388–1401. ACM, 2016.

[32] European Union. General data protection regulation. http://ec.europa.eu/justice/data-protection/reform/files/regulation_oj_en.pdf, April 2016.

[33] Export.gov. U.s.-eu & u.s.-swiss safe harbor frameworks, 2017.

[34] Marcelo Fantinato, Patrick CK Hung, Ying Jiang, Jorge Roa, Pablo Villarreal, Mohammed Melaisi, and Fernanda Amancio. A Survey on Purchase Intention of Hello Barbie in Brazil and Argentina. In *Computing in Smart Toys*, pages 21–34. Springer, 2017.

[35] Kassem Fawaz, Kyu-Han Kim, and Kang G. Shin. Protecting Privacy of BLE Device Users. In *USENIX Security Symposium*, pages 1205–1221, 2016.

[36] Campaign for a Commercial Free Childhood. Hell no barbie: 8 reasons to leave hello barbie on the shelf. http://www.commercialfreechildhood.org/action/hell-no-barbie-8-reasons-leave-hello-barbie-shelf, 2015.

[37] Forbrukerrådet. Investigation of privacy and security issues with smart toys. https://fil.forbrukerradet.no/wp-content/uploads/2016/12/2016-11-technical-analysis-of-the-dolls-bouvet.pdf, November 2016.

[38] Forbrukerrådet. #watchout: Analysis of smartwatches for children. https://fil.forbrukerradet.no/wp-content/uploads/2017/10/watchout-rapport-october-2017.pdf, October 2017.

[39] Michael C Grace, Wu Zhou, Xuxian Jiang, and Ahmad-Reza Sadeghi. Unsafe exposure analysis of mobile in-app advertisements. In *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*, pages 101–112. ACM, 2012.

[40] Stacy Gray. Always on: privacy implications of microphone-enabled devices. In *Future of privacy forum*, 2016.

[41] HackerOne. Toytalk: bug bounty program - get rewards through hackerone, 2017.

[42] Andrew Hilts, Christopher Parsons, and Jeffrey Knockel. Every step you fake: A comparative analysis of fitness tracker privacy and security, 2016.

[43] Candice Hoke, Lorrie Faith Cranor, Pedro Giovanni Leon, and Alyssa Au. Are they worth reading? an in-depth analysis of online trackers' privacy policies. *I/S: a journal of law and policy for the information society*, 2015.

[44] Donell Holloway and Lelia Green. The Internet of toys. *Communication Research and Practice*, 2(4):506–519, 2016.

[45] Leif-Erik Holtz, Katharina Nocun, and Marit Hansen. Towards displaying privacy information with icons. In *IFIP PrimeLife International Summer School on Privacy and Identity Management for Life*, 2010.

[46] Owen Hughes. VTech porn blunder leaves NSFW content easily accessible on children's tablet. https://uk.news.yahoo.com/vtech-porn-blunder-leaves-nsfw-125236717.html, December 2016.

[47] Patrick CK Hung, Farkhund Iqbal, Shih-Chia Huang, Mohammed Melaisi, and Kevin Pang. A glance of child's play privacy in smart toys. In *International Conference on Cloud Computing and Security*, pages 217–231. Springer, 2016.

[48] Troy Hunt. Data from connected cloudpets teddy bears leaked and ransomed, exposing kids' voice messages, Mar 2017.

[49] Troy Hunt. When children are breached – inside the massive VTech hack. https://www.troyhunt.com/when-children-are-breached-inside/, November 2017.

[50] JAMS. Jams mediation, arbitration, adr services. https://www.jamsadr.com/, 2017.

[51] Sławomir Jasek. Gattacking Bluetooth Smart Devices. http://gattack.io/whitepaper.pdf.

[52] Meg Leta Jones and Kevin Meurer. Can (and should) Hello Barbie keep a secret? In *IEEE International Symposium on Ethics in Engineering, Science and Technology (ETHICS)*, pages 1–6. IEEE, 2016.

[53] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W Reeder. A nutrition label for privacy. In *Symposium on Usable Privacy and Security (SOUPS'09)*, Mountain View, CA, USA, 2009.

[54] Deguang Kong, Lei Cen, and Hongxia Jin. Autoreb: Automatically understanding the review-to-behavior fidelity in android applications. In *Conference on Computer and Communications Security (CCS'15)*, 2015.

[55] Andreas Kurtz, Hugo Gascon, Tobias Becker, Konrad Rieck, and Felix Freiling. Fingerprinting mobile devices using personalized configurations. *Privacy Enhancing Technologies*, 2016.

[56] Maaaaz. Androwarn, Mar 2013.

[57] Moustafa Mahmoud, Md Zakir Hossen, Hesham Barakat, Mohammad Mannan, and Amr Youssef. Towards a comprehensive analytical framework for smart toy privacy practices. In *International Workshop on Socio-Technical Aspects in Security and Trust (STAST)*, 2017.

[58] Jonathan R Mayer and John C Mitchell. Third-party web tracking: Policy and technology. In *IEEE Symposium on Security and Privacy (SP)*, pages 413–427. IEEE, 2012.

[59] Emily McReynolds, Sarah Hubbard, Timothy Lau, Aditya Saraf, Maya Cakmak, and Franziska Roesner. Toys that listen: a study of parents, children, and internet-connected toys. In *ACM Conference on Human Factors in Computing Systems (CHI'17)*, Denver, CO, USA, 2017.

[60] Emily McReynolds, Sarah Hubbard, Timothy Lau, Aditya Saraf, Maya Cakmak, and Franziska Roesner. Toys that listen: A study of parents, children, and internet-connected toys. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 5197–5207. ACM, 2017.

[61] Alessio Merlo and Gabriel Claudiu Georgiu. Riskindroid: machine learning-based risk analysis on android. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, 2017.

[62] Anthony D Miyazaki, Andrea JS Stanaland, and May O Lwin. Self-regulatory safeguards and the online privacy of preteen children. *Journal of Advertising*, 38(4):79–91, 2009.

[63] Corinne Moini. Mandated ethical hacking-a repackaged solution. *Rich. JL & Tech.*, 23:1, 2016.

[64] Corinne Moini. Protecting Privacy in the Era of Smart Toys: Does Hello Barbie Have a Duty to Report. *Cath. UJL & Tech*, 25:281, 2016.

[65] Motherboard. One of the largest hacks yet exposes data on hundreds of thousands of kids. https://motherboard.vice.com/en_us/article/yp3z5v/one-of-the-largest-hacks-yet-exposes-data-on-hundreds-of-thousands-of-kids, 2015.

[66] Alexios Mylonas, Stelios Dritsas, Bill Tsoumas, and Dimitris Gritzalis. Smartphone security evaluation: The malware attack case. In *2011 Proceedings of the International Conference on Security and Cryptography (SECRYPT)*, pages 25–36. IEEE,

2011.

[67] Agnes Nairn and Alexander Dew. Pop-ups, pop-unders, banners and buttons: The ethics of online advertising to primary school children. *Journal of Direct, Data and Digital Marketing Practice*, 9(1):30–46, Jul 2007.

[68] Bill Nelson. Children's connected toys: data security and privacy concerns, Dec 2016.

[69] Bill Nelson. Children's connected toys: Data security and privacy concerns. https://www.billnelson.senate.gov/sites/default/files/12.14.16_Ranking_Member_Nelson_Report_on_Connected_Toys.pdf, December 2016.

[70] Philip Oltermann. German parents told to destroy doll that can spy on children, Feb 2017.

[71] OPC. Collecting from kids? Ten tips for services aimed at children and youth. https://www.priv.gc.ca/en/privacy-topics/privacy-and-kids/02_05_d_62_tips, December 2015.

[72] Paul Pearce, Adrienne Porter Felt, Gabriel Nunez, and David Wagner. Addroid: Privilege separation for applications and advertisers in android. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, pages 71–72. Acm, 2012.

[73] Pen Test Partners. VTech Innotab Max vulnerable to trivial data extraction. https://www.pentestpartners.com/security-blog/vtech-innotab-max-vulnerable-to-trivial-data-extraction/, December 2015.

[74] The Usable Privacy Policy Research Project. Usable privacy, 2016.

[75] Laura Rafferty, Patrick Hung, Marcelo Fantinato, Sarajane Marques Peres, Iqbal Farkhund, Sy-Yen Kuo, and Shih-Chia Huang. Towards a privacy rule conceptual model for smart toys. In *Hawaii International Conference on System Sciences*, 2017.

[76] Rapid7. HACKING IoT: A Case Study on Baby Monitor Exposures and Vulnerabilities. https://www.rapid7.com/docs/Hacking-IoT-A-Case-Study-on-Baby-Monitor-Exposures-and-Vulnerabilities.pdf, September 2015.

[77] Rapid7. R7-2015-27 and R7-2015-24: Fisher-Price Smart Toy hereO GPS Platform Vulnerabilities (FIXED). https://blog.rapid7.com/2016/02/02/security-vulnerabilities-within-fisher-price-smart-toy-hereo-gps-platform/, February 2016.

[78] Robert W Reeder, Patrick Gage Kelley, Aleecia M McDonald, and Lorrie Faith Cranor. A user study of the expandable grid applied to p3p privacy policy visualization. In *ACM Workshop on Privacy in the Electronic Society (WPES'08)*, Alexandria, VA, USA, 2008.

[79] Juniper Research. Smart toy sales to grow threefold to exceed $15.5 billion by 2022, 2017.

[80] Ivan Ristic. *Bulletproof SSL and TLS: Understanding and Deploying SSL/TLS and PKI to Secure Servers and Web Applications*. Feisty Duck, 2013.

[81] Norman Sadeh, Alessandro Acquisti, Travis D Breaux, Lorrie Faith Cranor, Aleecia M McDonald, Joel R Reidenberg, Noah A Smith, Fei Liu, N Cameron Russell, Florian Schaub, et al. The usable privacy policy project. Technical Report CMU-ISR-13-119, Carnegie Mellon University, 2013.

[82] Inc. Somerset Recon. Hello Barbie Initial Security Analysis, January 2016.

[83] Somerset Recon, Inc. Hello barbie security analysis. https://static1.squarespace.com/static/543effd8e4b095fba39dfe59/t/56a66d424bf1187ad34383b2/1453747529070/HelloBarbieSecurityAnalysis.pdf, January 2016.

[84] Emmeline Taylor and Katina Michael. Smart toys that are the stuff of nightmares. *IEEE Technology and Society Magazine*, 2016.

[85] Emmeline Taylor and Katina Michael. Smart toys that are the stuff of nightmares. *IEEE Technology and Society Magazine*, 35(1):8–10, 2016.

[86] TRUSTe. Submit a report - watchdog, 2017.

[87] United States Federal Bureau of Investigation. Public service announcement. https://www.ic3.gov/media/2017/170717.aspx, July 2017.

[88] Junia Valente and Alvaro A Cardenas. Security & Privacy in Smart Toys. 2017.

[89] VTech. FAQ about Cyber Attack on VTech Learning Lodge. https://www.vtech.com/en/press_release/2016/faq-about-cyber-attack-on-vtech-learning-lodge/, December 2016.

[90] World Wide Web Consortium (W3C). P3p: The platform for privacy preferences, nov 2007.

[91] Which? Safety alert: See how easy it is for almost anyone to hack your child's connected toys. https://www.which.co.uk/news/2017/11/safety-alert-see-how-easy-it-is-for-almost-anyone-to-hack-your-childs-connected-toys/, November 2017.

[92] Gang Xiong, Jiayin Tong, Ye Xu, Hongliang Yu, and Yong Zhao. A survey of network attacks based on protocol vulnerabilities. In *Asia-Pacific Web Conference*, pages 246–257. Springer, 2014.

[93] Benjamin Yankson, Farkhund Iqbal, and Patrick CK Hung. Privacy preservation framework for smart connected toys. In *Computing in Smart Toys*. 2017.

[94] Mu Zhang, Yue Duan, Qian Feng, and Heng Yin. Towards automatic generation of security-centric descriptions for android apps. In *ACM SIGSAC Conference on Computer and Communications Security (CCS'15)*, 2015.

[95] Sebastian Zimmeck, Ziqi Wang, Lieyong Zou, Roger Iyengar, Bin Liu, Florian Schaub, Shomir Wilson, Norman Sadeh, Steven M Bellovin, and Joel Reidenberg. Automated analysis of privacy requirements for mobile apps. In *Network and Distributed System Security Symposium (NDSS'17)*, San Diego, CA, USA, 2017.