Android Device Risk Assessment Tool:

Using Common Permissions to Identify Applications Used in Intimate Partner Violence

Kathryn Reardon

November 24, 2020

Abstract

* Write after finishing report
* 120 – 500 words, or 1-2 paragraphs
  + 25% on purpose/importance of research (Introduction)
  + 25% on what I did (Methods)
  + 35% on what I found (Results/Evaluation)
  + 15% on implications of research (Discussion)

1 Introduction

Intimate partner violence (IPV) is considered a public health concern in the United States by the CDC. This fact can be shocking for the uninformed whose idea of a public health concern is heart disease or a viral pandemic, but the four abusive behaviors of physical violence, sexual violence, stalking, and psychological aggression can lead to injury and death as assuredly as any disease [1]. The statistics differ slightly depending on the source, but approximately 1 in 4 women and 1 in 10 men have reported experiencing some form of IPV. One method that abusers use to control, manipulate, and harm their victims is surveillance spyware installed on their victim’s devices. This topic has only recently been researched, but it is a security issue of great importance because the abusers do not have to be technologically savvy to employ their attacks. Additionally, many victims are uninformed about the abuser’s tech capabilities and do not know how to deal with it even once the surveillance is suspected or discovered.

This project aims to continue the burgeoning research and security solutions begun by students and faculty primarily at Cornell Tech and New York University [2, 3, 4, 5, 6, 7]. The work done by Sam Havron et al. in creating a clinical computer security procedure and IPV Spyware Discovery tool (ISDi) is the major inspiration for the project [5]. Further work by Kevin Roundy et al. also provided ideas during the research phase [7]. I had several goals for the project, the first of which was to improve ISDi’s efficacy with a sort of “signature”-based spyware detection, as ISDi relies on blacklisting. The second goal was to obtain the permissions of known Android spyware applications and analyze them for commonalities in a guilt-by-association approach where any permission that was frequently used in spyware was more likely to indicate that an unknown app was also spyware. The third goal was to inspect victim devices without arousing suspicion of the attacker. The fourth goal was to programmatically assess the likelihood of an app being spyware with accuracy, as false negatives can be dangerous for the client (victim) while false positives obfuscate the danger. The final goal was to do everything in a way that is easy for a client to understand, leading to informed decision making regarding their device.

To accomplish these goals, I have designed three separate pieces, two support tools and a graphic user interface to display the findings. The first tool, AnalyzeAndroidPermissions, is Java code used to read the permissions from the AndroidManifest.xml files of a group of sample applications. It categorizes and sorts them according to Android protection level, number of times used, and whether they are present in spyware or popular, non-spyware apps. The second tool, Horoscope, is an Android application that on its face appears to be a simple daily horoscope. However, when the app is launched, it also gathers installation data of the apps on the device and saves them to a file to be used with the third tool. This final tool, AssessAppRisk, is a Java Swing application which lists every installed app with the data taken from the Horoscope app, and also displays a risk value decided by heuristic weights, a risk assessment, and descriptions of the permissions’ capabilities. AssessAppRisk includes a whitelist of 17 of the most popular apps on the Google Play Store, so the risk assessment ranges from whitelisted, very unlikely, unlikely, slightly likely, likely, and very likely to be spyware. If these tools were to be used in a real situation, it would follow the example of the field study introducing ISDi: a technician works with the client to run the tools on their device and go over the results with them.

Since IPV has evolved into a computer security problem, it is important to create a threat model that contextualizes the roles of attacker and victim and answers the questions of what the attacker knows, has access to, and can do. We must have a model both for the initial attack on the victim device and in the use of project’s tool. In the first case, these questions have been answered by the previous researchers [3]. As an aside, we use the terms attacker and victim here in the sense of an adversary model, though they align with the role of abuser and the target of abuse. “Victim” is not meant to be a slight or implication toward any person who is a target of or survivor of abuse. As for who the attacker and victim are, they tend to be intimate partners—spouses, boyfriends or girlfriends, exes, etc. There are other types of relationships, such as that of parent and child (where either party may play either role depending on age and circumstances) or even platonic friendships [8], but the relationships described in anecdotes by clients of IPV studies were only those of a romantic partner, ex-partner, and/or parent of their child(ren). The attacker and victim may live together, have lived together in the past, or never together at all. This is an important distinction for this threat model; while there are insider attacks wherein the attacker uses their organizational status to carry out an attack on that organization, it is easier to imagine a hacker thousands of miles away carrying out a remote attack. This is also not an attack on some faceless corporation, but one on a specific individual, and one whom the attacker knows very well.

What the attacker knows in this situation is both much and very little. On the social engineering side, the attacker may know many secrets that allow them to compromise the victim’s devices or accounts. Over the course of their relationship with the victim, they may come to know or compel the victim to tell them answers to security questions, like favorite color or birthday; they can watch or compel the victim to type passwords and PINs. On the other hand, it has been found that these attackers are not technically sophisticated, and Freed et al. go so far as to term them “UI-bound adversaries,” as they employ their attacks through a standard user interface with which they can be authenticated, or download the applications examined in this project to do their surveillance [3].

Another drastic difference from the typical adversary model is the attacker’s access to the victim device since most, if not all, assume the attacker’s target is someone else’s device [11]. One commonly reported scenario in cases of IPV is the device or its service is bought or paid for by the attacker. This gives the attacker control in innumerable ways, with such examples as the attacker confiscating or destroying the device, controlling associated digital accounts such as the mobile family plan or iCloud, and even manipulating relationships by giving a child a device with the intention to harass the adult target through it. Physical access to the device is key, especially since many of the spyware apps marketed towards these attackers require it.

Finally, what the attacker can do varies and has changed over time. In the past, an attacker could search for simple terms on the Google Play store like “track my girlfriend’s phone without them knowing” or “read SMS from another phone” and found many apps to choose from [4]. Following the warnings of security researchers, Google has removed many spyware apps from its store and filtered out IPV-related search terms, and it seems Android also has made changes to its APIs to make certain features of spyware apps unusable. Still, there were and still are apps that can be found from a Google search, and it is simple to disable a device’s protections, e.g., Google’s Play Protect, to install such off-store apps. One particularly nasty app called Cerberus boasts of uninstall protection, remote wipe, lock with password, blocking the power menu, and those are only the capabilities that prevent the victim from reclaiming their privacy. Apps claim they can track the device’s location, take pictures, record video and audio, forward text messages, read deleted messages, and practically any other type of privacy breach one can imagine. This makes them powerful and scary tools indeed. On the other hand, some apps are not as they appear. Some apps tested for this project did not work past an introductory screen, and others triggered anti-virus software which flagged the app as a phishing attack. The Zscaler research term confirmed another case when analyzing the code of the keylogger app SPYMIE, finding a hard-coded email address with a timer to send surveilled data every minute [12]. Additionally, while some apps are free to download, most can only be used after purchasing a subscription plan.

I had to evaluate a threat model when designing my project as well. Many aspects from the previous threat model remain. I kept many design considerations from Sam Havron et al. when they were creating ISDi. First we had to consider if the attacker would know that a spyware scanning tool was being used on the victim device, and the following consideration was what they would do upon learning of the tool’s use or the victim’s participation in a security clinic. I will discuss the implications of these considerations on both projects in later sections.

Given that the main goal of my project is to identify Android spyware used in IPV, I would evaluate my project’s success by the accuracy with which it does so. I would like to see AssessAppRisk flag 75% of known spyware apps as some degree of likely to be spyware while flagging only 25% of non-spyware apps (false positives). I will show an evaluation in a following section.

2 Background and Related Work

The groundwork for this project was laid by an interdisciplinary group of researchers at Cornell Tech, Cornell University, and New York University in a series of studies beginning in 2017 and continuing today. The first study largely identifies IPV as a security issue through interviews with 40 IPV professionals and nine focus groups with 32 survivors of IPV. These interviews revealed how abusers use technology, what clients and professionals understand about said technology, how professionals advise clients about technology, and how the law understands technology used in IPV [2]. The next paper was another qualitative study with 89 participants to detail exactly how abusers use technology in IPV, which discovered technologically unsophisticated methods, such as social engineering and downloaded applications. Most importantly, they began to consider how to mitigate these types of attacks [3]. This led to an investigation of spyware used in intimate partner surveillance which revealed both spyware and what they termed dual-use apps which have legitimate uses but can be exploited by abusers, such as Find My Friends or anti-theft applications. In this study, they also found that existing anti-virus and anti-spyware tools did not consistently catch dual-use applications [4].

The next study, mentioned earlier, produced a consultation service for IPV victims. In this service, a trained technologist worked with the client and an IPV professional to answer a standardized technology assessment questionnaire, create a diagram summarizing the client’s digital assets, manually check the client’s device for security configurations, and scan the device using ISDi [5]. ISDi is a Python application that does not require installation on the client’s device. This is to avoid notifying the abuser that the device is being investigated, since some spyware applications keep track of application installations or take screenshots while the device is being used. Instead, the technologist uses a USB connection to a laptop where ISDi is run in a browser. The researchers tested a USB connection on devices with six more capable spyware apps installed and were reasonably confident that the apps would not pick up on ISDi. ISDi’s major limitation is that its spyware detection is rather simple, using a blacklist of application names that were found through machine learning [4].

The next paper published in 2019 examined the findings of the clinical computer security approach and was more focused on sociology than computer science, reiterating some of the ideas from previous papers about the security needs and understandings of IPV victims [6]. The last study used the spyware applications unearthed by Chatterjee et al. as a seed set to discover “creepware,” apps similar to the surveillance-based spyware previously discussed but used more generally for interpersonal attacks. This paper is as important to this project as the paper concerning ISDi as it provided a method for spyware detection beyond blacklisting. The researchers developed an algorithm called CreepRank which uses the principle of guilt-by-association to identify applications as creepware [7].

3 Dataset Description and Properties

* Describe apps used for analysis and how/why they were obtained/chosen
* How were permissions analyzed
* Sorting permissions by protection level

4 Using Guilt-by-Association

* Brief description of CreepRank algorithm
* Permissions grouped by usage
* Deciding which permissions are guilty of spyware use
* Heuristic weights and rationale

5 The Risk Assessment Tool

* How the Horoscope app and risk assessment tool are intended to be used

1. The Horoscope app

* How it works
* Why it is designed the way it is
* What it provides for the main tool

1. The Risk Assessment tool

* How it works
* What it shows
* How results should be interpreted

6 Evaluating Efficacy

* Show testing examples
* Evaluate results—satisfactory? Does it meet established goals? Mixed results?
* Thoughts about why results are such

7 Discussion

* Did I achieve my goals? Which ones?
* Limitations/room for improvement
* Implications of research
* Future work/context of project beyond this experimental stage

8 Conclusion

* Summarize accomplishments

References

Appendix

Code

Screenshots

REFERENCES

1. CDC. 2020. Intimate Partner Violence. Retrieved November 17, 2020 from

https://www.cdc.gov/violenceprevention/intimatepartnerviolence/index.html.

1. Diana Freed, Jackeline Palmer, Diana Elizabeth Minchala, Karen Levy, Thomas

Ristenpart, and Nicola Dell. 2017. Digital technologies and intimate partner violence: A qualitative analysis with multiple stakeholders. *Proceedings of the ACM on Human-Computer Interaction* 1, CSCW, Article 46 (November 2017), 22 pages. DOI:https://doi.org/10.1145/3134681

1. Diana Freed, Jackeline Palmer, Diana Minchala, Karen Levy, Thomas Ristenpart, and

Nicola Dell. 2018. “A stalker's paradise”: How intimate partner abusers exploit technology. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (*CHI '18*). April 21 – 26, 2018, Montréal, Canada. Association for Computing Machinery, New York, NY, USA, Paper 667, 1–13. DOI:https://doi.org/10.1145/3173574.3174241

1. Rahul Chatterjee et al. 2018. The spyware used in intimate partner violence. In

*Proceedings of the 2018 IEEE Symposium on Security and Privacy* (*SP*), May 21 – 23, 2018, San Francisco, California, USA. 441-458. DOI:https://doi.ieeecomputersociety.org/ 10.1109/SP.2018.00061

1. Sam Havron, Diana Freed, Rahul Chatterjee, Damon McCoy, Nicola Dell, and Thomas

Ristenpart. 2019. Clinical computer security for victims of intimate partner violence. In *Proceedings of the 28th USENIX Security Symposium* (*Security 2019*). August 14 – 16, 2019, Santa Clara, California, USA. 105-122. DOI:https://dl.acm.org/doi/10.5555/3361338.3361347

1. Diana Freed, Sam Havron, Emily Tseng, Andrea Gallardo, Rahul Chatterjee, Thomas

Ristenpart, and Nicola Dell. 2019. “Is my phone hacked?” Analyzing clinical computer security interventions with survivors of intimate partner violence. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW, Article 202 (November 2019), 24 pages. DOI:https://doi.org/10.1145/3359304

1. Kevin A. Roundy, Paula Barmaimon Mendelberg, Nicola Dell, Damon McCoy, Daniel

Nissani, Thomas Ristenpart, and Acar Tamersoy. 2020. The many kinds of creepware used for interpersonal attacks. In *Proceedings of the 41st IEEE Symposium on Security and Privacy* (*Oakland 2020*). May 18 – 20, 2020, Online. 626-643. DOI: https://doi.ieeecomputersociety.org/10.1109/SP40000.2020.00069

1. Karen Levy, Bruce Schneier, 2020. Privacy threats in intimate relationships. *Journal of*

*Cybersecurity* 6, 1 (May 2020), 13 pages. DOI:https://doi.org/10.1093/cybsec/tyaa006.

1. Nithya Vasudevan. How to Get List of Installed Apps in Android. Retrieved November

17, 2020 from https://theopentutorials.com/tutorials/android/listview/how-to-get-list-of-installed-apps-in-android.

1. Oracle. How to Write a List Selection Listener. Retrieved November 17, 2020 from

https://docs.oracle.com/javase/tutorial/uiswing/events/listselectionlistener.html.

1. Quang Do, Ben Martini, and Kim-Kwang Raymond Choo. 2019. The role of the

adversary in applied security research. *Computers & Security*, 81 (March 2019), 156-181. DOI:https://doi.org/10.1016/j.cose.2018.12.002.

1. Shivang Desai. Why You Shouldn’t Trust “Safe” Spying Apps! Retrieved November

17, 2020 from https://www.zscaler.com/blogs/security-research/why-you-shouldnt-trust-

safe-spying-apps.

1. Computer Security and Privacy for Survivors of Intimate Partner Violence. Retrieved

November 17, 2020 from https://www.ipvtechresearch.org/.