Poster: Android App Forensic Evidence Database

Chen Shi, Chris Cheng, Mitchell Kerr, Connor Kocolowski, Emmett Kozlowski
Matthew Lawlor, Jacob Stair, Neil Gong, and Yong Guan
Department of Electrical and Computer Engineering
Iowa State University, Ames, Iowa, USA 50014

I. Introduction

A recent study of global available apps, AndroZoo [1], shows that the number of various real-world apps has exceeded 8 million and is still rapidly growing. In contrast, commercial mobile device forensic tools such as Cellebrite UEFD [2] supports the profiles of about 6,000 apps, may not function sufficiently well in assisting real-world digital forensic investigations. Therefore, investigators oftentimes have to use manual forensic investigation approaches, when mobile devices had apps installed but not listed in the 6,000 apps supported by Cellebrite UEFD. However, such manual investigations heavily rely on investigator's experience, knowledge and skills, which are often both time-consuming and error-prone. For example, a recent investigation on a 5-year-old Nexus 7 tablet with 90 installed apps found to have more than 20,000 files imaged from the device. It is often impractical to manually inspect such massive number of files one by one and maintain the required level of quality and error guarantees such that investigators can generate trustworthy legal report within certain required time bounds, which may in turn leads to even more serious completeness and quality problems.

To tackle the challenges in mobile device forensics, we are building an ultimately the largest Android App Forensic Evidence Database (AndroidAED) using the two techniques developed by our group, namely *EviHunter*, with the goal of mitigating the vulnerabilities and reliability concerns of digital forensic investigations. Through using AndroidAED, digital forensic practitioners can simply query the database to find all the possible evidence data (e.g., locations, photos, call logs, time, etc.) generated by the app, where they are (e.g., evidentiary file path), and what types of evidence data. Moreover, consider that the apps installed on the suspect's device can be varied because of app's version, source of installation (app store), AndroidAED hosts the apps (APKs, including most recent and past versions of the same app) collected from various app markets: Google Play [3] and over 50 other app stores. For each collected APK (Application Package) file, we apply both static EviHunter [4] and dynamic app analysis tool [5] developed by us, to parse the evidentiary file paths and the carried evidence data types. The further development of the static and dynamic app analysis tools are on-going to handle complex situations caused by obfuscation, naive code, and 3rd party libraries. The main contributions of AndroidAED are summarized as follows:

1) AndroidAED, to the best of our knowledge, is the first Android app forensic evidence database with highest precision and coverage in discovering evidence generated from apps, collected from major app stores.
2) AndroidAED will be made open for public access that significantly improves the investigation of evidence from mobile devices, and other security and privacy research on mobile apps (malware/ransomware, stego-detection, visual media manipulation (DARPA MediFor)).
3) AndroidAED will be kept updating to provide the most up-to-date evidentiary data for real-world apps ranging from very popular to very unpopular ones available from app stores across the world.

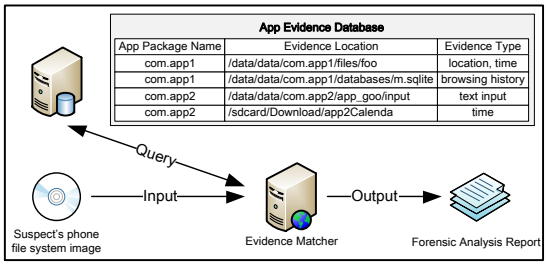

Fig. 1. A typical usage scenario of AndroidAED for Digital Forensic Investigation

II. Android App Evidence Database Usage Scenarios

An automatic forensic analysis approach leveraging AndroidAED is presented at Fig. 1. The procedure starts from the physical acquisition of suspect's mobile device, and thereafter the extraction of the file system image. Once obtaining the imaged evidence via Cellebrite or other forensic tools, through parsing the installed apps information on device, the digital forensic practitioner can obtain the evidentiary data from AndroidAED for all the apps installed on that suspect's device. Using the query results and the evidence types of interest, a simple matcher can retrieve the files that have their paths matched. However, when it comes to the dynamic path issue, addressed in EviHunter [4], such as the file path contains the string value of timestamp, a regular expression matching is required additionally to model the file path.

In addition to the evidentiary file matching, AndroidAED provides the functionality of bulk query, which allows the user to query a large number of results satisfying given constraints,

This work was funded by the Center for Statistics and Applications in Forensic Evidence (CSAFE) through Cooperative Agreement #70NANB15H176 between NIST and Iowa State University. Contact: Yong Guan, guan@iastate.edu

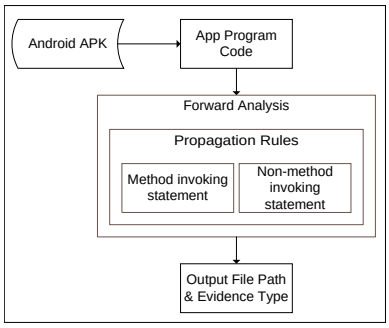

Fig. 2. Overview of EviHunter Workflow.

Forensic Android App Database

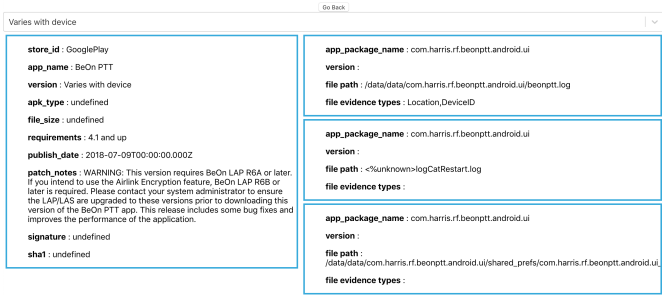

Fig. 3. Single record of the app BeOn PTT.

such as the range of published dates, developer, permission requirements. Not only the evidentiary data, but also original APK file, metadata and user comments are included in the database, making academic study feasible. With the signature (hash value) of each APK recorded, more newly-defined/found forensic analysis result can be updated in the future.

III. Android App Evidence Database Specifications

The more apps we can include in the database, the more false-negatives we can reduce or avoid in real-world mobile evidence investigations. In order to provide a more complete coverage of real-world cases, we have developed app crawlers for different app markets. At the current stage, we have completed seven crawlers and 40 more are under development. The current supported app markets include: Google Play Store [3], APKPure [6], Uptodown [7], APKMirror [8], Aptoide [9] and F-Droid [10]. For each app (distinguished by its package name), each version's APK and its corresponding metadata was archived as single record in our database. Apps' metadata include app category, description, developer, reviews, published date, version number, file size, permission requirements, minimum system API, and the number of installation. Additionally, we retrieve the signature file from APK so as to build the profile and avoid duplication caused by different sources of downloads. In case of the fabricated signature file, another pair of signature was generated by our own to mitigate the potential errors.

We are running a large scale of app collections from various app stores with the support of our Senior Design Project teams. As a prototype demo, our AndroidAED has 5,323 apps and 45,745 versions of APKs archived. After acquiring APK file, two program analysis tools are leveraged to analyze possible evidentiary files and evidence types. Static program analysis tool, EviHunter [4], was deployed to model the call graph and perform codebase analysis to generate the file path and its carried evidence data type. The workflow of EviHunter is presented at Fig. 2, after extracting program code from the given input APK file, it performs forward analysis for each statement in control flow graph. Through applying the propagation rules for the visited statements, EviHunter computes the propagation of variables' carried evidence types and string value of file path. Once a sink method, defined as the methods writing data to the file system, EviHunter reports the constructed file path and the merged evidence types from the given position of variable(s). Because of the existing challenges of defining evidence types, we currently support "location", "time", "visited URLs" and "text input" these four types. More types can be added once being clearly defined. Dynamic program analysis tool [5] was utilized to generate the evidentiary data result, but in different ways. It starts from installing the APK file on mobile device, and then a script was programmed to randomly click the items on the app. Once a tainted data was detected in the file system, its corresponding evidence type and the actual file path was reported. Through parsing APK file by these tools, evidentiary data can be reported and updated into AndroidAED accordingly. An example of the app's profile including the forensic analysis result is shown in Fig. 3.

IV. Summary and Future Directions

Our goal is to build the database AndroidAED that ultimately becomes the largest Android app forensic evidence database in the world. We are working with crime labs and digital forensic communities, and expect AndroidAED will have large societal impacts in significantly improve the efficiency and reliability of real-world digital forensic casework. We also keep all the older versions of apps as digital forensic archives and continue updating the apps and evidence results into the database everyday. Additionally, we will look into possible ways of providing in-memory forensic evidence from memory snapshots or program state reconstruction. We envision that AndroidAED and apps in it will provide a strong base for future security, privacy, and forensics research.

References

[1] K. Allix, T. F. Bissyandé, J. Klein, and Y. Le Traon, "Androzoo: Collecting millions of android apps for the research community," ser. ACM MSR, 2016.
[2] "Cellebrite ufed ultimate," 2019. [Online]. Available: https://www.cellebrite.com/en/products/ufed-ultimate/
[3] "Google play store," 2019. [Online]. Available: https://play.google.com/store
[4] C. C.-C. Cheng, C. Shi, N. Z. Gong, and Y. Guan, "Evihunter: Identifying digital evidence in the permanent storage of android devices via static analysis," in *ACM CCS*, 2018.
[5] Z. Xu, C. Shi, C. C.-C. Cheng, N. Z. Gong, and Y. Guan, "A dynamic taint analysis tool for android app forensics," in *SADFE*, 2018.
[6] "Apkpure," 2019. [Online]. Available: https://apkpure.com/
[7] "Uptodown," 2019. [Online]. Available: https://en.uptodown.com/
[8] "Apkmirror," 2019. [Online]. Available: https://www.apkmirror.com/
[9] "Aptoide," 2019. [Online]. Available: https://en.aptoide.com/
[10] "F-droid," 2019. [Online]. Available: https://f-droid.org/en/packages/