Novel Anti-forensics Approaches for Smart Phones

S. Azadegan, W. Yu, H. Liu, M. Sistani, S. Acharya
Towson University
{sazadegan, wyu, sacharya}@towson.edu  {hliu3, msista1}@students.towson.edu

Abstract

Anti-forensics is defined as a method undertaken to thwart the digital investigation process conducted by forensic investigators. In this paper we introduce the design and implementation of three novel anti-forensic approaches for data deletion and manipulation on three of the market-leading forensics tools. The evaluation results of our proposed anti-forensic approaches on a typical Android cell phone device demonstrate the severe limitations of current forensic tools and conclude the critical need for modifications and/or redesigns of these contemporary forensic tools to maintain accuracy and legitimacy of forensic data acquisition and analysis.

1. Introduction

There has been an exponential growth in mobile phones usage for the past several years and this trend is expected to continue for the foreseeable future. Smart phones with increasing storage capacity and functionality have outnumbered the use of other computing devices and are becoming the primary choice for personal communication among the world population. A large amount of information from phone book to photo albums and videos, to emails and text messages, to financial records and GPS records, is stored on these mobile devices. Without doubt, the use of mobile phones is becoming an irreplaceable terminal device for a vast majority of Internet users.

Unfortunately, with the growth and popularity of cell phones usage, there has been a similar increase in the use of such devices for conducting digital crimes. Recent statistics indicate that with the increase in mobile phone subscriber numbers, over five billion users [1, 2, 3], there is a similar increase in the use of mobile devices for criminal activities and the confiscation of these devices at crime scenes. Additionally, law enforcement officials have warned of the increase in the digital crime rates with the recent introductions of 3G and 4G technologies [4]. Because legal experts use the data extracted from these confiscated devices and present them as evidence in the court, this has made the area of mobile phone forensics a critical and integral part of law enforcement operations and justice system around the world.

Due to the pervasive usage of these mobile devices, recovering evidence from these devices become an integrated part of digital forensics and also leads to greater challenges [5, 6]. The authors in [5] have identified six general categories of challenges: (i) carriers and manufacturers, (ii) data preservation, (iii) power and data connectors, (iv) operating systems and communication protocols, (v) security mechanisms, and (vi) unique data formats, which all can be attributed to the lack of standardization and homogeneity in cell phones. Cell phones are continuously modified and redesigned as existing technologies improve and newer technologies are introduced. Due to this, today’s forensic science experts operate in what is called the forensic tool spiral [7]. Newer and more powerful versions of forensic tools are created regularly to address the new features and functionalities of the fast evolving phones and their technologies. This has lead to the task of data acquisition and analysis [8, 9, 10] from these devices extremely challenging. In addition, in today’s digital world there is a continuous dependancy of law enforcement officials on the evidence and inferences provided by these forensic tools to assist in the investigation of crimes and towards the acquisition of critical data provided to judicial systems. This has further lead to the critical need for the validation of the approaches used and evidences provided by these forensic tools.

Furthermore, today’s forensic industry is facing a critical challenge of numerous anti-forensic approaches and applications that have been designed and used by hackers and hacking applications to cause the loss of authenticity and legitimacy of the result provided by the forensic tools [11]. In this regard, the term Anti-forensics [12, 13] pertains to the methods undertaken in order to thwart the digital investigation process conducted by legitimate forensic investigators. Forensic science researchers today use the method of anti-forensics as an adversarial approach in order to understand and infer...
the pattern and behavior of a hacker or hacking tool and/or application. Such an approach is important to help understand the design steps necessary to establish authenticity and legitimacy of the use of the various forensic tools and applications.

In recent times researchers from both academia and industry have developed various applications for conducting anti-forensics. These approaches can be broadly classified as Data Hiding, Artifact Wiping, Trail Obfuscation, and Attacks on the individual forensic tools [14]. Data hiding can be accomplished in a variety of ways. Hiding writing, also known as steganography [15], has been around for a long time. There are numerous ways of hiding data from normal search paths. Data can be hidden in slack or unallocated portions on memory locations, scattered throughout memory, as well as placed as metadata of many types of files, etc. Artifact wiping tools are programs that wipe and destroy data files using multiple overwrites that make any form of data retrieval impossible to use [16]. Trial obfuscation is usually accomplished by wiping and/or altering log files and/or system event files or altering the timestamps of various files, etc [17]. The anonymous nature of execution and use of these tools makes the task of the forensic expert practically impossible. Direct attacks on the forensic tool operation are amongst the newest forms of anti-forensics. These attacks can be launched at various execution steps of the tool and lead to inaccuracy and/or loss of evidence and hindrance to digital investigation.

Hence, it is critical to understand the design and operation of modern forensic tools and evaluate the various approaches by which the authenticity and legitimacy of these tools can be compromised. To this end, in this paper we focus on the design and implementation of three novel anti-forensic approaches, which tamper with data acquisition process by disrupting the forensic devices connection to smart phones, completely deleting the contact list data, to partially deleting the data marked as sensitive, to completely replacing the data with fake data. These applications were implemented and tested on three of the market-leading forensic tools using a typical Android mobile device. Our evaluation demonstrates the limitations of current forensic tools and concludes the critical need for modifications and/or redesigns of these contemporary forensic tools.

The remainder of the paper is organized as follows. In Section 2, we present a brief overview of the related work in this area. In Section 3, we review the forensic tools. In Section 4, we present the discussion of three different anti-forensic approaches and the evaluation results on contemporary forensic tools for data deletion/manipulation for an Android cell phone device. Finally, in Section 5, we present the conclusion and outline our future research directions.

2. Related Work

Cell phone forensics aims at acquiring and analyzing data in cellular phones. Forensics tools for cell phones are quite different from those for personal computers. The challenges of cell phone forensics have been shown in [5, 6, 18]. In particular, Garfinkel [6] outlined the current forensic research directions and argued that to move forward the community needs to adopt standardized, modular approaches for data representation and forensic processing.

There have been a number of efforts for evaluating forensics tools and developing forensics techniques for cell phones in the past [19, 20, 21, 22, 23, 24]. For example, Curran et al. presented mobile phone forensic analysis, what it means, who avails of it and the software tools used [19]. Somasheker et al. presented the overview of tools available for examining PDAs and cell phones [20]. Ting et al. investigated the dynamic behavior of the mobile phone’s volatile memory and presented an automated system to perform a live memory forensic analysis for mobile phones [22]. Mokhonoana et al. presented the forensics technique, which uses an on-phone forensic tool to collect the contents of the device and store it on removable storage [23]. Connor presented the forensics techniques based on the analysis of call detail records and the corresponding tower-antenna pairs, which can provide useful information as evidence in a criminal trial [24].

Anti-forensics (AF) is the set of tactics and measures taken by someone who wants to thwart the digital investigation process [11, 12, 13]. In particular, Kessler [14] describes some AF tools and methods, under the broad classifications of data hiding, artifact wiping, trail obfuscation, and attacks on the forensics tools themselves. Distefano et al. also discussed the classification of the classification of the anti-forensics techniques (e.g., destroying evidence, hiding evidence, eliminating evidence sources, and counterfeiting evidence) [25]. Garfinkel categorizes the traditional anti-forensics technique, including the encrypted file systems and disk sanitization utilities, and presents a survey of recent anti-forensics tools, including Timestomp and Transmogrify, along with the strategies for detection and countermeasures against anti-forensics [26].

There are other researches related to anti-forensics. For example, Foster et al. developed Timestomp, a utility to allow for the deletion or
modification of time stamp-related information on files [17], network traffic can be anonymized via encryption and use of intermediaries to protect the content and communication relationship such as Anonymizer and Tor [27, 28], watermarking, steganography and covert channels are the information hiding techniques which can encompass applications, such as copyright protection for digital media [29, 30, 3, 32]. FreeOTFE [33] is an open source program to create and mount virtual encrypted disk on Windows and Linux. It provides two levels of plausible deniability, on the fly encryption and supports numerous encryption algorithms.

Different from the existing research, our work is the first one, which enables the protection of privacy of phone’s data and the design and implementation of three novel anti-forensic approaches for data deletion and manipulation on a representative forensic tool, using a typical Android mobile device.

3. Forensic Tools

There are three primary methods of acquiring data using forensic tools: (i) Manual Acquisition, (ii) Physical Acquisition, and (iii) Logical Acquisition [33], which will described below.

In Manual acquisition, the user interface can be utilized to investigate the content of the memory. The device is used as normal and pictures are taken from the screen. This method has the advantage that the operating system makes the transformation of raw data into human interpretable format. The disadvantage of this method is that only data visible to the operating system can be recovered and that all data are only available in form of pictures. In contrast, Physical acquisition implies a bit-by-bit copy of an entire physical storage. This acquisition method has the advantage of allowing deleted files to be examined. Physical extraction acquires information from the device by direct access to the flash memories. Generally, this is harder to achieve because the device vendors needs to secure against arbitrary reading of memory so that a device may be locked to a certain operator. Finally, Logical acquisition implies a bit-by-bit copy of logical storage objects that reside on a logical store. This method of acquisition has the advantage that system data structures are easier for a tool to extract and organize. Logical extraction acquires information from the device using the interface for synchronizing the contents of the phone with the analyzing device (e.g., PC). This method usually does not produce any deleted information, due to it normally being removed from the file system of the phone.

We started this project by assessing the reliability and accuracy of the five market-leading commercial forensics tools [34] using the evaluation standards provided by NIST [35, 36]. In November 2010, researchers at NIST also provided a thorough evaluation of four of these tools [37, 38, 39, 40]. In the following, we will only provide a brief description of three forensics tools and the phone used those tools to test the proposed anti-forensics applications.

(i) Paraben Device Seizure [41] is an advanced forensic acquisition and analysis tool for examining cell phones, PDAs, and GPS devices. The tool is able to acquire both logical and physical data. It contains a report generation tool that allows for the convenient presentation of data. The tool is able to generate reports using the report wizard in csv, html, text or xls formats. The tool is designed to be able to recover the deleted data and retrieve physical data from some devices and has a fairly simple user interface.

(ii) XRY [42] is developed by Micro Systemation (MSAB), and based on our evaluation [34] is one of the best-dedicated mobile device forensic tools. ‘XRY Complete’ is a package containing both software and hardware to allow both logical and physical analysis of mobile devices. The unified logical/physical extraction wizard in XRY and the resulting reports help to show the examiner the full contents of the device in a clean and professional manner. The tool is able to connect to the cell phone via IR, Bluetooth or cable interfaces.

(iii) SecureView [43] tool provides various smart features for cell phone forensics such as: svSmart (ability for streamlining and presetting conditions to attain evidence in the field quickly); svPin (ability for unlocking CDMA cell phone passwords); svLoader (ability to download, analyze, verify and validate other sources and aid in creating csv files and upload, back up files from RIM and iPhone); and svReviewer (ability to share data without multiple licenses). The tool contains a SIM card reader to extract data from the SIM cards of GSM phones. The tool acquires data via cable, Bluetooth or IR interfaces. Reports are then generated in a print ready format. It provides a friendly interface and strong phone support for over 2000 phones. The tool also provides the option of generating a detail report of data and output in a pdf format. In addition, the report can be merged with the import report from other forensic tools by using Sxprobe.

In this project, we have used Motorola Droid 2.2 phone [44], which is an Android-based smart phone. This phone is based on Android 2.2 (Froyo) and
support CDMA 1X 800/1900 and EVDO with rich features.

4. Anti-Forensic Approaches

Harris [45] defines Anti-forensics (AF) as “any attempt to compromise the availability or usefulness of evidence in the forensic process”. In this section, we present three different Android anti-forensics applications. Each application either destroys or manipulates the contact list data on the phone, as soon as it detects that the phone has been connected to a forensics tool. The applications were implemented and tested with SecureView, Paraben and XRY forensics tools, which are briefly described in Section 3.

As stated in [33], we started our study by monitoring the communication between the tools and the cell phone. We used USBlyzer [46] for this purpose, which is the protocol analyzer software and it captures, decodes and displays communication of forensics tools going through USB device stack. Figure 1 depicts a snapshot of USBlyzer for the SecureView forensics tool. One of the fields not clearly shown in Figure 1 is a timestamp field for each transaction.

![Figure 1: Snapshot of USBlyzer](image)

Through this process, we recognized that all forensics tools follow a similar pattern of activities, listed below, to retrieve the data from the phone:

- Copy the application package to /data/local/tmp/ folder;
- Install the application;
- Delete the package file from the temporary folder;
- Start the application;
- Retrieve and transfer the data in the phone to the tool; and
- Uninstall the application after the completion of data retrieval the tool.

Table 1 captures the corresponding USBlyzer commands for the above steps performed by each tool (notice that some of the intermediate steps have been removed).

<table>
<thead>
<tr>
<th>Paraben</th>
<th>SecureView</th>
<th>XRY</th>
</tr>
</thead>
<tbody>
<tr>
<td>STAT</td>
<td>STAT</td>
<td>STAT</td>
</tr>
<tr>
<td>…../data/local/tmp</td>
<td>…/data/local/tmp</td>
<td>…/data/local/tmp</td>
</tr>
<tr>
<td>DataPilotTool2.apk</td>
<td>DataPilotTool2.apk</td>
<td>HelloAndroid.apk</td>
</tr>
<tr>
<td>shell:pm install /data/local/tmp/AndroidService.apk</td>
<td>shell:pm install /data/local/tmp/DataPilotTool2.apk</td>
<td>shell:pm install /data/local/tmp/HelloAndroid.apk</td>
</tr>
<tr>
<td>success..</td>
<td>Success..</td>
<td>Success..</td>
</tr>
<tr>
<td>Broadcasting: Intent {act=StartSeizureService,. Broadcast completed: result=0.</td>
<td>Starting: Intent {cmp=com.susteen.android.dptool2/ DataPilotTool2,.</td>
<td></td>
</tr>
<tr>
<td>&lt;request&gt;.</td>
<td>&lt;request&gt;.</td>
<td>&lt;request&gt;.</td>
</tr>
<tr>
<td>&lt;request&gt;.</td>
<td></td>
<td>WRTE.......</td>
</tr>
<tr>
<td>&lt;request&gt;.</td>
<td></td>
<td>...t...”&lt;&lt;&lt;</td>
</tr>
</tbody>
</table>
Comparing the timestamps of these activities, we noticed that there was a multi-second gap between the start of sending the file and start of retrieving information. Our proposed AF applications will take advantage of this period.

To avoid the overhead of running and monitoring all the USB transactions, we decided to monitor the Android logs, which also capture all the activities of the forensics tools. The application uses logcat command to read the log messages line by line and look for the initial connection of the forensic tool. This application running as a background service on the phone does not take up much resource. Table 2 lists the relevant logcat commands.

### Table 2: LogCat activities for SecureView

<table>
<thead>
<tr>
<th>Timestamp</th>
<th>Logcat Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>05-13 11:41:01.566</td>
<td>I/PackageParser( 1238): com.susteen.android.dptool2: compat added android.permission.WRITE_EXTERNAL_STORAGE</td>
</tr>
<tr>
<td>05-13 11:41:01.613</td>
<td>D/PackageManager( 1238): Scanning package com.susteen.android.dptool2</td>
</tr>
<tr>
<td>05-13 11:41:01.613</td>
<td>I/PackageManager( 1238): /data/app/com.susteen.android.dptool2-1.apk changed; unpacking</td>
</tr>
<tr>
<td>05-13 11:41:02.496</td>
<td>I/installld( 1140): move /data/dalvik-cache/data@<a href="mailto:app@com.susteen.android.dptool2-1.apk">app@com.susteen.android.dptool2-1.apk</a>@classes.dex -&gt; /data/dalvik-cache/data@<a href="mailto:app@com.susteen.android.dptool2-1.apk">app@com.susteen.android.dptool2-1.apk</a>@classes.dex</td>
</tr>
<tr>
<td>05-13 11:41:02.496</td>
<td>D/PackageManager( 1238): New package installed in /data/app/com.susteen.android.dptool2-1.apk</td>
</tr>
<tr>
<td>05-13 11:41:04.348</td>
<td>I/ActivityManager( 1238): Starting activity: Intent { flg=0x10000000 cmp=com.susteen.android.dptool2/.DataPilotTool2 }</td>
</tr>
<tr>
<td>05-13 11:41:04.980</td>
<td>V/ContactServer( 2674): starting RequestHandler thread.....</td>
</tr>
<tr>
<td>05-13 11:41:04.980</td>
<td>V/ContactDAO( 2674): content url:content://com.android.contacts/contacts</td>
</tr>
</tbody>
</table>

We now describe three AF approaches that can be used to compromise the data extraction process.

#### Sudden Death

This is the simplest approach and was discussed in details in [33]. The application is installed on the phone and configured to auto start after each boot up as a background service monitoring the phone logs. As soon as it detects that phone is connected to a forensic tool, it shuts the phone off and wipes out the data stored in the phone. The forensic tool which had a successful initial connection to the phone is forced to close its connection as a result. The disadvantage of this approach is that the detectives will clearly notice that the data retrieval process has been compromised and may use forensic tools such as XRY, which provide physical data acquisition to retrieve the data.

#### Erase Sensitive Data

In this approach, we assume that the sensitive data on the contact list has been already marked by the phone owner (e.g., savvy criminal, unfaithful spouse), in anticipation of the day that the data on the phone would be extracted by a forensics tool. In the current implementation, the application deletes any
entry on the contact list that starts with an asterisk, ‘*’. Similar to the sudden death approach the application monitors the log and as soon as it detects an initial communication from a forensics tool, it deletes all the entries in the contact list database marked by the owner. Since the application locks the database, it prevents the forensics tool from accessing the data. After sanitizing the data, the tool successfully retrieves the remaining data.

We tested the application for different contact list size (the largest with 1500 entries) and the extra time needed by the application was negligible. Table 3 shows the logcat entries’ timestamps and total required time for the retrieval of 200 contacts from the phone. The contact list was populated with random data and as many entries as half were marked sensitive. In the first case where the AF application was not running, it took 2297ms to establish the initial connection and 49807ms to retrieve 200 contacts from the phone. In the second case with the AF application running, it took 3875ms to establish the initial connection and remove the sensitive data (96 contacts), and 19771ms to retrieve the remaining 104 records.

The advantage of this method is that the data acquisition process by the tool will not be interrupted and the tool successfully retrieves the sanitized data with a minimal delay, though the data on the phone is tampered.

Table 3: Logcat entries for XRY forensics tool retrieving 200 contacts

<table>
<thead>
<tr>
<th>Without Anti-Forensics Application</th>
<th>Retrieval</th>
<th>06-14 14:44:35.284 I/PackageParser(1241): Start proc example.helloandroid for activity example.helloandroid/.HelloAndroid: pid=2696 uid=10097 gids={3003, 1015}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial connection</td>
<td>Start Data Retrieval</td>
<td>06-14 14:33:35.81 I/ActivityManager(1241): Start proc example.helloandroid for activity example.helloandroid/.HelloAndroid: pid=2336 uid=10097 gids={3003, 1015}</td>
</tr>
<tr>
<td>End Data retrieval</td>
<td>End Data retrieval</td>
<td>06-14 14:44:25.683 I/ActivityManager(1241): Displayed activity example.helloandroid/.HelloAndroid: 52104 ms</td>
</tr>
<tr>
<td>With Anti-Forensics Application</td>
<td></td>
<td>06-14 14:54:20.456 I/PackageParser(1241): example.helloandroid: compat added android.permission.WRITE_EXTERNAL_STORAGE</td>
</tr>
</tbody>
</table>

|            |            | 06-14 14:54:24.331 |

Replace All Data

In this approach, which is the most general one, the application upon detection of the forensics tool connection will replace all the original data in the contact list with fake data generated randomly by the application. The application once installed on the phone randomly generates a set of names from popular first and last names, phone numbers with valid area codes and gmail or yahoo email addresses containing the first and/or last name of the contact with additional numbers. This data will be used to update the contact list when the phone is connected to a forensic tool. Like the previous approach it makes it very hard at the beginning to notice that something has gone wrong. Of course, it is always possible to get a copy of the call list from the Internet Service Provider; however, other contact information such as name, email, address and phone numbers, etc. will be erased. Moreover, in most criminal cases, prepaid phone or SIM cards are used, which makes the call logs not very helpful in the investigation process.

Though these three applications only dealt with protecting the privacy of the contact list content, they demonstrated how easily one can block forensics tools from retrieving data from the phone. We are currently working on how to extend these applications to protect other data stored on the phone.

5. Conclusion

Mobile devices with increased storage capacity and wide range of features and functionalities have evolved into full-fledge computing platforms and provide an abundance of information ranging from contact lists and text messages to video files and GPS location information. Increasingly, the data extracted from these devices, using forensic tools, provide crucial evidence in both civil and criminal investigation. For this reason, an array of forensic tools has been developed to retrieve data from these devices so that it can aid in forensic investigations.

Anti-forensics, as a relatively new and growing discipline, is attracting a lot of attention as an attempt...
to compromise the availability or usefulness of evidence in the forensic process. In this paper, we presented three anti-forensics Android applications, which tamper with data acquisition process by completely deleting the contact list data, to partially deleting the data marked as sensitive, to completely replacing the data with fake data.

Through these applications developed in the paper show that how easily one can compromise potentially crucial evidences in an investigation process. As a double sword, we can use this type of applications for securing smart phones and protecting classified or highly sensitive data from falling into wrong hands. As ongoing work, we are working to extend these applications to protect other data stored on the phone and investigate possible countermeasures, such as randomizing the initial communication pattern to make it harder for the applications to block or interfere with the data extraction process.

Acknowledgements

This work was supported in part by the National Science Foundation (NSF) under SFS grant award 0416706. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References


