A Comparison of Forensic Acquisition Techniques for Android Devices: A case study investigation of Orweb browsing sessions

Nedaa Baker Al Barghouthy
Department of Electrical and Computer Engineering
College of Engineering
University of Sharjah
Sharjah, United Arab Emirates
Email: nedaa@sharjah.ac.ae

Andrew Marrington
Advanced Cyber Forensics Laboratory
College of Technological Innovation
Zayed University
Dubai, United Arab Emirates
Email: marrington@computer.org

***Abstract*—The issue of whether to "root" a small scale digital device in order to be able to execute acquisition tools with kernel-level privileges is a vexing one. In the early research literature about Android forensics, and in the commercial forensic tools alike, the common wisdom was that "rooting" the device modified its memory only minimally, and enabled more complete acquisition of digital evidence, and thus was, on balance, an acceptable procedure. This wisdom has been subsequently challenged, and alternative approaches to complete acquisition without "rooting" the device have been proposed. In this work, we address the issue of forensic acquisition techniques for Android devices through a case study we conducted to reconstruct browser sessions carried out using the Orweb private web browser. Orweb is an Android browser which uses Onion Routing to anonymize web traffic, and which records no browsing history. Physical and logical examinations were performed on both rooted and non-rooted Samsung Galaxy S2 smartphones running Android 4.1.1. The results indicate that for investigations of Orweb browsing history, there is no advantage to rooting the device. We conclude that, at least for similar investigations, rooting the device is unnecessary and thus should be avoided.**

***Index Terms*—Orweb; Android; acquisition; root; rooting.**

I. Introduction

Small scale digital devices pose numerous challenges for digital investigations. Their ubiquity is surpassed only by their hetereogenity when compared to personal computers, and because of their hetereogenity, there is a great diversity in both the tools and techniques which investigators need to employ to retrieve evidence from small scale digital devices. If we consider computer hard disk forensics, an investigator needs only a handful of cables to support different hard disk bus types, and possibly different write-blockers, and the same software tools and processes will support image acquisition from all the different types of hard disk deployed in desktop computers for the past twenty years. By comparison, in small scale digital device forensics, an investigator may need dozens of cables, different write-blockers (if they are available at all), different software tools and different processes just to be able to acquire images of mobile phones and tablets manufactured over the past twenty months! Worse still, because of the comparatively immature nature of small scale digital device forensics compared to computer forensics, the "best practice" procedures for forensic acquisition and analysis of small scale digital devices are not always clear to practitioners or researchers due to competing vendor claims and unresolved issues in the literature. To illustrate this point, consider the forensic soundness of installing a (presumably tested) rootkit on an Android device in order to physically acquire the device's flash storage. "Rooting" the device naturally modifies it, potentially overwriting some data of evidentiary value, and for this reason, it is discouraged by some authors [1]. Nevertheless, many commercial tools require the Android device to be "rooted" before physical acquisition is possible [2]. In this work we seek to address this ambiguity.

This paper considers the issue of the efficacy different acquisition techniques for Android devices through the medium of a case study investigation. The case involved private web browsing using the Orweb browser app. Orweb is a browser which uses Onion Routing to anonymize web traffic, and which records no browsing history [3]. A full discussion of Tor and its operations is beyond the scope of this work, but can be found in [4] - suffice it to say that Orweb is the "official" Tor browser for Android, and used with Orbot, facilitates anonymized web browsing and bypasses Internet filtering. There were two motivations for attempting to reconstruct Orweb browser sessions:

1) Determining an effective method for investigating potential malicious use of the Orweb browser to conceal illegal browser activity.
2) Evaluating the claimed privacy benefits of using Orweb by determining whether traces of Orweb browser sessions were left on the device.

In addressing these motivations, we performed three different types of acquisition on the Android device to retrieve Orweb browser session traces. The techniques we performed are detailed in section IV, and they are based on the techniques employed in the related work we discuss earlier in section II.

978-1-4799-3223-8/14/$31.00 ©2014 IEEE

Our results, in section V, detail all the traces of our simulated Orweb browser sessions uncovered using each acquisition methdology, and show that at least in comparable cases, there is no advantage to rooting an Android device over the recovery partition approach. Finally, in section VI we conclude that rooting Android devices to facilitate physical acquisition is no longer justified, and discuss future work.

II. Related Work

Small scale digital devices have a number of important differences from computers from the perspective of digital investigations. Whereas most computers have user-removable hard disk drives, the storage media of small scale digital devices is varied in its configuration and usually not easily physically accessible. Most small scale digital devices use some type of flash memory, but there is greater diversity in operating systems and filesystems than exists in the personal computer world. Consequently acquisition of storage media in small scale digital device forensics is a good deal more difficult than acquisition of storage media in computer forensics. This difficulty is reflected in the variety of acquisition techniques reported on in the scientific literature.

Lessard and Kessler described the acquisition of Android-based devices using a rootkit and the Android Debug Bridge (ADB) to open a root-privilege shell on the Android device, from which a trusted 'dd' binary can be executed to acquire an image of the device's memory (both internal and removable) [2]. In this work we refer to this approach, which is employed by many commercial mobile phone forensics tools, as the rootkit method. Installing a rootkit on the device necessarily modifies the device, although such modification may be very minor. In order to avoid modifying the user data partition, where they assume the highest value digital evidence will generally reside, Vidas et al. propose reflashing the device's recovery partition and replacing it with a forensic acquisition environment [1]. The device is rebooted into "recovery mode" and then an image is acquired through the trusted forensic acquisition environment. In this work we refer to this technique as the "recovery mode" method.

Both the rootkit method and the recovery mode method are aimed to facilitate the acquisition of the Android device's "secondary storage" - the equivalent of the hard disk in computer forensics. However, as in computer forensics, the Android device's RAM is also a potentially rich source of digital evidence. It may contain encryption keys, passwords, and evidence of running malware, among many other types of evidence. Sylve et al. proposed an approach for acquiring an image of RAM from an Android device and then analysing that image using LiME [5]. In this work, a rootkit must still be applied to the device in order to facilitate live memory acquisition. This means that the image of RAM acquired from the device will include the "footprint" of the rootkit. Further, since the device will be running at the time of acquisition, the image acquired is more a "smear" of the RAM over a brief period of time than a snapshot of the RAM at an atomic moment in time (rather like a rapidly moving object appears blurred in a photograph taken by a camera with a low shutter speed).

In previous work, the authors have examined the artifacts left behind on a Samsung Galaxy S2 Android mobile device by private browsing sessions using the Orweb Tor Browser as compared to the regular browser [6]. Orweb is an Android browser which incorporates a Tor proxy to allow for anonymized web browsing on an Android smartphone or tablet computer [3]. In this previous work, we were unable to find traces of Orweb browser activity without rooting the device. A logical backup was performed on the device both before and after rooting, and traces of the Orweb browsing activity were found in the post-rootkit logical image only. We did not employ the "recovery mode" method in this previous work, and our acquisitions were logical only.

III. Contribution

As noted above, Vidas et al. proposed reflashing the Android device's recovery partition with forensic acquisition software, and then rebooting the device into recovery mode and physically acquiring an image of the device's flash memory as an alternative to rooting the Android device in order to facilitate the acquisition of the image of the same memory [1]. Vidas et al. did not evaluate the "recovery mode" method experimentally. In this work, using a simulated case, we show that the "recovery mode" method is just as effective as rooting the device measured in terms of traces of the suspicious browser activity retrieved.

IV. Methodology

In order to address the research questions mentioned above, we designed an experiment. We conducted a sample browsing session using Orweb. The device was subsequently examined twice, once prior to rooting using the "recovery mode" method, and once afterwards. In each examination we searched for evidence of the users browsing activity. Our hypothesis was that the "recovery mode" method would be just as effective as the rootkit method as Vidas et al. believed [1]. If, in our experiment, the examination of the image acquired via the "recovery mode" method discovers the same evidence, or more evidence, when compared to the image acquired via the rootkit method, then the hypothesis is sustained and the "recovery mode" method should be preferred over the rootkit method for the acquisition of the Android device's flash memory. Otherwise, if the rootkit method facilitates the discovery of additional evidence, then our hypothesis is disproven.

A. Instruments

We employed the following instruments in our experiment:

- Laptop PC running Windows 7 Enterprise SP1 as the forensics workstation
- Samsung Galaxy S2 smartphone running Android 4.0.3
- Orweb v2.28 Android app (includes Orbot)
- Recovery Clockwork v4.0.1.5 image for Galaxy S2 to be loaded into the smartphone which allows reboot into ClockWorkMod (CWM) Recovery mode.

- Unyaffs tool for Windows to extract the device backup image files.
- Samsung Kies v2, Odin3 and S2 Root software for rooting the smartphone
- SQLite Database Browser v1.2
- CF-Root kernel, to be installed on the smartphone
- Android Debug Bridge software for the forensics workstation
- Micro USB cable to connect the forensics workstation to the smartphone
- New SD card to be inserted into the smartphone
- FTK Imager for acquiring the forensic image of the external memory card of the smartphone
- FTK Toolkit v1.7
- Tableau Write Blocker to prevent modification of the suspects SD card during image acquisition

B. Usage Scenario

On the Android smartphone, we performed the following usage scenario to substitute for a real crime under investigation:

- We visited the Facebook website (www.facebook.com not the Android app).
- We logged into Facebook website as "Butti Hamad a user profile created specifically for this experiment.
- We started a Facebook Message instant messaging conversation with a Facebook Friend, the user "Hind Rashid (also a user profile created specifically for this experiment).
- As Butti Hamad, we sent an image photo file to Hind Rashid.

Although the user profiles and image file used in this experiment were innocuous, this simple scenario was designed to be deliberately similar, in terms of the actions performed, to a cyber-bullying or harassment scenario.

C. Recovery Mode Acquisition Method

We copied a Galaxy S2 recovery image (the ClockWorkMod Recovery image) onto the blank SD card, and swapped this blank SD card for the removable SD card inside the Android smartphone. The device's original removable SD card was plugged into the Tableau write-blocker and physically acquired using FTK Imager, leaving only the internal memory to acquire.

We rebooted the device into the standard recovery mode by holding the volume up, home button and power switch of the device. We then selected to "Update from external card" and selected our downloaded image in order to reflash the recovery partition. The external SD card was then removed, wiped, and replaced. We then rebooted the device into the ClockWorkMod Recovery mode, and performed a backup with Nandroid, backing up the internal memory card's partitions as image files. The device was then connected to the forensic workstation, the Android Debug Bridge was started, and used to copy the image files to the external SD card using the "adb pull" command. After the copying process was completed, the device was turned off, and the external SD card was removed. We finally plugged the external SD card into the Tableau write-blocker and copied the acquired image files to the forensic workstation for examination. MD5 hashes were generated both at the time of initial backup by the Nandroid utility, and at the time of copying the files from the external SD card to the forensic workstation.

D. Rootkit Acquisition Method

We placed the original removable external SD card back into the Galaxy S2 device, and restarted the device. We then enabled USB debugging on the Android device, and powered it down again. The device was then powered up into downloading mode (by holding the volume down button, power button and menu button on the Galaxy S2 device), and connected the device to the forensic workstation via the USB cable. We ran Odin3 on the forensic workstation, loaded the new kernel on the device and rebooted, applying the rootkit on restart through the forensic workstation. This completed the "rooting" of the Galaxy S2 device.

We then removed the device's external SD card and inserted it into the Tableau write-blocker. Using the forensic workstation plugged into the Tableau device, we physically acquired the external SD card using FTK Imager, leaving only the internal memory to acquire. The internal memory was acquired with Nandroid, and the generated image files were copied from the device to the forensic workstation. MD5 hashes were generated by the Nandroid backup process and these were compared to MD5 hash values of the files copied to the forensic workstation.

E. Analysis

The Samsung Galaxy S2 Android smartphone device uses the YAFFS2 filesystem. Although later versions of the FTK Toolkit are able to mount YAFFS2 partitions, our version of FTK (v1.71) does not support YAFFS2. Even this old version of FTK has the ability to keyword search the binary image file, and to data carve files out of the images despite not being able to interpret the filesystem itself.

In order to examine the acquired images at a logical level, we used the "unyaffs" utility to extract files from the image files obtained from the Samsung S2 device through both acquisition methods. We also employed the SQLite Database Browser to interpret SQLite files recovered from the images using the unyaffs utility. We were able to compliment our analysis of the logical filesystem by file carving and keyword searching from the raw binary images using FTK, although in this experiment all the evidence we located was found through the examination of the logical filesystem through unyaffs.

V. Results

We located the same evidence on both sets of images - those obtained using the "recovery mode" method and those obtained using the rootkit method. From the "data" images from both the rooted and unrooted devices, we recovered the "info.guardianproject.browser" directory with its contents,

TABLE I
SUMMARY OF EVIDENCE LOCATED IN EACH SET OF IMAGES

Evidence Description	**Rootkit Method Images**	**Recovery Mode Images**
History of visited URLs	Yes	Yes
Facebook account login and password	Login name found only	Login name found only
Participant Facebook IDs	Yes	Yes
Participant email ID	Yes	Yes
Chat conversation	Ciphertext only	Ciphertext only
Chat date and timestamp	Yes	Yes
File transferred between participants	Yes (path only)	Yes (path only)
GPS coordinates of the participants	No	No
RSA public key	Yes	Yes
Accepted port numbers to be used while browsing in Orweb	Yes	Yes
Rejected port numbers to be ignored by Orbot	Yes	Yes

including the browser cache for the Orweb browser, from which we could obtain the public key of the Onion routing circuitry and accepted/rejected ports. As reported in section IV, our hypothesis was that the same evidence would be found using the "recovery mode" method as could be found using the rootkit method. This hypothesis was sustained. In table I, we report the items of evidence we recovered about the Facebook via Orweb scenario we described in section IV-B.

Although we were able to retrieve the Facebook login name on rooted device, we were not able to retrieve the password. In a real-world law enforcement investigation into Facebook activity, this information could be obtained from Facebook through an appropriate court order.

VI. CONCLUSION

We conclude that since reflashing the recovery partition with acquisition software does not modify the user or system partitions, but rooting does, and both are otherwise equally effective for facilitating physical acquisition of the Android's flash memory as shown by our results in section V, it is preferable not to root the device. In drawing this conclusion, we must note the limitation that this work only examines the issue of Android acquisition techniques through the prism of reconstructing private browsing sessions in Orweb.

It is also worth discussing the different contexts in which a forensic investigator may seek to root the device. While our results indicate that the "recovery mode" method and rootkit method are equally effective as means to facilitate the physical acquisition of the Android device's secondary memory (i.e. its flash storage), our results do not apply to the acquisition of live memory - the Android device's RAM. There may still, therefore, be a legitimate reason to root Android devices for digital forensic purposes if the investigator wishes to acquire an image of the device's RAM.

We share the opinion of Vidas et al. that "rooting" an Android device to facilitate physical acquisition is not ideal since it modifies, however slightly, the device [1]. Therefore, as an avenue of future work, we propose to look into the use of alternative acquisition techniques to rooting for the purposes of acquiring an Android device's RAM. The "recovery mode" method is unlikely to be a suitable acquisition method for the forensic acquisition of RAM contents since rebooting the device will naturally be very disruptive to RAM contents. This is especially true since the device will probably need to be rebooted at least twice, first to reflash the recovery partition, and second to actually boot into the recovery image, as in our procedure described in section IV-C. The rootkit method is not necessarily unacceptable, as observed by Sylve et al., it is quite normal for desktop PCs for some sort of root privilege escalation to be required to dump memory [5]. Nevertheless, if it can be avoided, it should, and this seems a worthy topic for future work.

REFERENCES

[1] T. Vidas, C. Zhang, and N. Christin, "Toward a general collection methodology for Android devices," *Digital Investigation*, vol. 8, pp. S14–S24, Aug. 2011.

[2] J. Lessard and G. Kessler, "Android Forensics: Simplifying Cell Phone Examinations," *Small Scale Digital Device Forensics Journal*, vol. 4, no. 1, 2010.

[3] The Guardian Project, "Orweb: Proxy+Privacy Browser," 2013. [Online]. Available: https://guardianproject.info/apps/orweb/

[4] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," Naval Research Lab, Washington D.C., Tech. Rep., 2004.

[5] J. Sylve, A. Case, L. Marziale, and G. G. Richard, "Acquisition and analysis of volatile memory from android devices," *Digital Investigation*, vol. 8, no. 3-4, pp. 175–184, 2012.

[6] N. Al Barghouthy, A. Marrington, and I. Baggili, "The forensic investigation of android private browsing sessions using orweb," in *5th International Conference on CSIT*. IEEE Computer Society, 2013.