BlackBerry PlayBook Backup Forensic Analysis

Mohamed Al Marzougy, Ibrahim Baggili, and Andrew Marrington

Advanced Cyber Forensics Research Laboratory, College of Technological Innovation,
Zayed University, Abu Dhabi, U.A.E.
Mohamed.Almarzougy@gmail.com, {Ibrahim.Baggili, Andrew.Marrington}@zu.ac.ae

Abstract. Due to the numerous complicating factors in the field of small scale digital device forensics, physical acquisition of the storage of such devices is often not possible (at least not without destroying the device). As an alternative, forensic examiners often gather digital evidence from small scale digital devices through logical acquisition. This paper focuses on analyzing the backup file generated for the BlackBerry PlayBook device, using the BlackBerry Desktop Management software to perform the logical acquisition. Our work involved analyzing the generated ".bbb" file looking for traces and artifacts of user activity on the device. Our results identified key files that can assist in creating a profile of the device's usage. Information about BlackBerry smart phone devices connected to the tablet was also recovered.

Keywords: BlackBerry, Forensics, PlayBook, Backup.

1 Introduction

The BlackBerry PlayBook is Research in Motion's (RIM) entrant into the heated tablet race which includes the iPad and various Android tablets. One of the main differences between the PlayBook device and other tablets is the ability to tether (via Bluetooth) to a BlackBerry smart phone for network access while away from WiFi networks at home or in the office, as compared to using an on-board 3G modem for that purpose. This tethering is provided by the BlackBerry Bridge feature that extends the functionality of the paired BlackBerry smart phone to the PlayBook's larger screen, enabling the viewing of emails, messages and files stored on the phone.

Although the iPad and the various Android tablets run a tablet-version of an operating system designed for a smart phone, the BlackBerry PlayBook runs a custom operating system. This means that research into the forensic acquisition of BlackBerry smart phones may not be applicable to the PlayBook device. To date, there has been no research performed on the forensic acquisition and analysis of the PlayBook's backup structure. Although the PlayBook has a comparatively small market-share [1], the PlayBook was the first tablet to gain FIPS 140-2 certification and cleared to be used by the U.S. Government [2]. Therefore, it is a worthwhile exercise to study the forensic acquisition, analysis and examination of the device via its backup structure. This approach has recently been applied successfully to the iPad [3] and we therefore thought to investigate its applicability to the BlackBerry PlayBook.

The remainder of this paper is organized as follows: in section 2 we briefly discuss the literature about the forensic examination of various types of tablet computers. In section 3, we describe the methodology for our experiment and we discuss our findings in section 4. In section 5 we draw conclusions from our work and we finish by discussing future research work into this area of small scale digital device forensics.

2 Background

Mobile phones and tablets are of particular interest to forensic investigations for the simple reason that due to their mobility they are likely to be in regular contact with suspects and/or victims throughout the course of the events under investigation. With enormous diversity in operating system software, hardware specifications, and vendors, small-scale digital devices like smart phones and tablets are an area of serious concern in digital forensic research [4].

Small-scale digital device forensics is a rapidly evolving subfield of digital forensics. The initial popularity of the iPhone and subsequently the iPad led to research into the retrieval and analysis of digital evidence from these devices [5][6][7]. There has been some research into Android devices [8], although it has been almost exclusively focused on phones and much remains to be done before a generalized methodology for Android forensics is possible [9]. There has also been some research on BlackBerry smart phone devices [10], but at the time of writing there is little published research about the BlackBerry PlayBook tablet, which is the focus of this paper.

2.1 iPhone and iPad

The iPhone, iPod Touch, and iPad all run the iOS operating system, and may be conceived of as broadly similar devices from a forensics perspective. All iOS devices interface with a personal computer or accessory peripherals through a proprietary port on the bottom of the device which connects to the computer's Universal Serial Bus (USB) port via a special cable. None of the iOS devices feature removable storage and consequently, any digital forensic examination of the device must take place via this cable.

Physical acquisition for iOS devices is limited to commercial products and law enforcement personnel. Andrew Hoog and Katie Strzempka [11] reviewed most tools that support iOS device forensics using the criteria: installation, acquisition, reporting and accuracy, where they came up with a ranking system they used to rank 13 digital forensics products and methodologies. The Zdziarski method scored the highest (4.1) where the rest averaged 3.3. Zdziarski's iPhone forensics method is one of the few which does not require the target device to be jailbroken - all an examiner has to do is put the device into recovery mode and load Zdziarski's tool into the device's RAM. The technique is conceptually similar to using a boot CD – essentially the device boots to an "alternate" system partition that has all the necessary software to run a "dd" command and create a forensic image of the user partition, bypassing any password protection. The National Institute of Standards and Technology validated the Zdziarski method as forensically sound [12].

Gómez-Miralles and Arnedo-Moreno employ jailbreaking in their approach, which uses the Apple Camera Connection Kit for the iPad to connect the device to an external hard drive [13]. After the iPad is jailbroken, OpenSSH and core utilities (coreutils) are installed on it, and the investigator connects to the device from a computer on the same WiFi network as the iPad using ssh. The "dd" command is issued to the target device specifying that the output is to be stored on the external hard drive connected via the camera connection kit.

2.2 Android Devices

Similar to the iPhone, Android keeps all the system files and some of the user information protected on the kernel level. Consequently, many forensic scientists suggest that the device should be "rooted" (a similar process to jailbreaking) to facilitate examination [9]. The Android file system is "Yet Another Flash File System 2" (YAFFS2). YAFFS, developed in 2002, was the first file system designed for NAND (Not-AND) flash memory devices. YAFFS2 was designed in 2004 in response to the availability of larger sized NAND flash devices; older chips support a 512 byte page size whereas newer NAND memory has 2096 byte pages. YAFFS2 is backward compatible with YAFFS [8].

The first and most obvious step is to perform a traditional forensics analysis of the microSD card from the Android device. This step will obviously only result in the acquisition of whatever data has been stored to the SD card but not the data which is stored in the device's non-removable memory. Android device SD cards use the FAT32 file system and are easily imaged and examined using traditional forensics tools (including write-blocking hardware).

In order to acquire access to the Android device's internal memory as opposed to simply the SD card, USB debugging must be enabled on the device. This mode can be enabled by the user through the appropriate configuration menu on the Android device. If the Android device's keylock is active, then the investigator requires the user's passcode to gain access to the configuration menu. According to Lessard and Kessler [8], unless USB debugging has been enabled, it is not possible to root the Android device. Golubev [14] explains that in the absence of the passcode, root access is necessary to bypass the Android device's keylock. This creates a "chicken and the egg" scenario where if the keylock passcode is unknown, the investigator must disable the keylock remotely via root access, but if the investigator cannot disable the keylock, he/she will be unable to root the Android device.

The exact process of rooting an Android device varies depending on the hardware manufacturer follows the same general process. This process requires inserting an SD card (preferably fresh and not the one used by the device as it may store evidence on it) and enabling USB debugging mode, then, through the use of Android Development Tools (ADT, part of the Android SDK) and the Android Development Bridge (ADB), a small program is copied to the SD card. This program is usually copied to /data/local/tmp, a folder where most installation files reside. The program is then run in order to root the device.

Other research has focused upon the analysis of the live memory of Android devices. Researchers have developed a tool that performs a dump of each running process' memory [15]. Although excellent for the analysis of a single process (such as a single running application), many other potentially interesting parts of the Android device's memory are not analyzed including in-kernel structures, networking information, etc. Another issue is that this approach requires memory to be extracted separately for each process of interest, which requires a number of interactions with the live system, increasing the chance that evidence will be contaminated. Sylve et al developed a kernel module that can be loaded to a rooted Android device to dump the memory of the device to the device's SD card with very high accuracy [16].

2.3 BlackBerry Devices

BlackBerry devices have long had the reputation for security, both with respect to the data stored on the device and to the security of emails and messages sent to and from the device. Previous work studying BlackBerry smart phone devices found that data was only forensically recoverable on devices where the users had not employed the device's encryption features [10]. However, the BlackBerry PlayBook uses a different operating system entirely from the BlackBerry OS used on the generations of BlackBerry smart phones up to this point. The BlackBerry Tablet OS is based on QNX Neutrino, an OS that is employed to run on many other portable devices. This operating system is Unix-based and features a microkernel.

BlackBerry devices were among the first smart devices to hit the market and as a result they became popular among government officials and corporate customers alike. Most BlackBerry devices come with the option to completely encrypt its memory. Further, the device makes it possible to encrypt the device's Secure Digital (SD) card as well. It is also possible to wipe a BlackBerry device remotely in the event that the device has been lost or stolen. BlackBerry devices, both the BlackBerry smart phones and the BlackBerry PlayBook, can also be backed up to a desktop computer using the BlackBerry Desktop Manager software. These backups may contain much information of forensic value to an investigator, just as they do for the iPhone [5] and iPad [3].

3 Methodology

Our method can be summarized as using a BlackBerry PlayBook device under manual observation, involving recording of all actions and their outcomes, before backing the device up with BlackBerry Desktop Manager and then analyzing the backup files produced to determine those of most potential interest to an investigator and their structure.

3.1 Test Equipment

Hardware:

- 64 GB BlackBerry PlayBook running OS 2.0.7971
- BlackBerry Bold 9900 running OS 7.1 Bundle 921 (7.1.0.267, Platform 5.1.0.230)
- IBM ThinkVantage with 2.6 Ghz Quad Core Intel processor, 4 GB RAM running Windows XP Professional, Service Pack 3.

Software and tools:

- BlackBerry Desktop Software 6.1.0.35
- Facebook for BB PlayBook 2.2.1.7
- WinRAR 3.30
- Hex Workshop 6.6
- SQLite Browser 2.0b1
- AccessData FTK 3.2
- Snagit

3.2 Test Procedure

The BlackBerry PlayBook device was initiated and connected to a wireless network as part of the initiation process. The device was connected to the lab's wireless network and the timezone was selected. After that the device required a BlackBerry ID, which was created using the following details:

- BlackBerry ID: bbpbmail@gmail.com
- First name: ZUPlayBook
- Last name: Student
- BlackBerry ID username: bbpbmail@gmail.com
- Password: zuBlackBerry
- Recovery Question: Where?
- Recovery Answer: Here
- Screen name: ZU

After the successful BlackBerry ID registration, the device was forced to update to OS 2.0.7971 and went through the first launch tutorials and demo. After that, the device was connected to the BlackBerry 9900 smart phone through the BlackBerry Bridge connection (over Bluetooth). Accessing the BlackBerry Bridge applications required the smart phone's password. The PlayBook then accessed emails from the smart phone through the bridge to the first author's email address, and we sent and received test emails to and from the account bbpbmail@gmail.com. The next bridge app we used was the BlackBerry Messenger (BBM), specifically checking received messages and then sending and receiving some BBM messages to members of the smart phone's contact list. We then disconnected the PlayBook from BlackBerry Bridge.

The next step was using a new feature in OS 2.0: direct email setup. Using this feature, the PlayBook device is used to directly receive and send emails over WiFi without the need for a tethered smart phone device connected via BlackBerry Bridge.

Subsequent to that, we performed some browsing activities on the PlayBook device, using the default browser, and then we started to run the YouTube and FaceBook applications. We also used the camera to take two photos and one video. Finally, a hotspot was created using the BlackBerry smart phone, and the PlayBook device was connected to that hotspot.

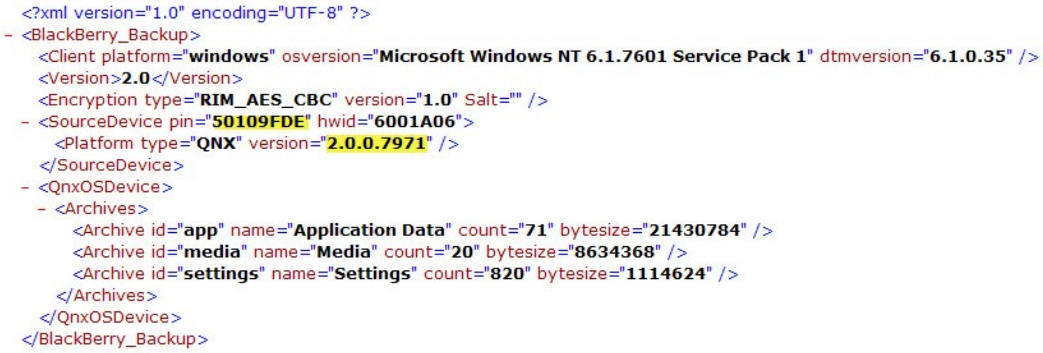
After that the device was connected to the PC to capture a backup. From the Desktop Software the backup option was set to “Full (all device data and settings)”.

After the backup was taken, more operations were made for comparison. One of the image files was deleted, a website was deleted from the browsing history and more images were copied to the device using the Desktop Manager Software. Files named ***dizer.jpg*** and ***low.jpg*** and ***chub.jpg*** were copied using the file explorer of the Desktop Manager Software. Then from the device the file ***chub.jpg*** was deleted. The device was then backed up again.

WinRAR was used to extract the files from the .bbb files, which are ZIP files with the “bbb” extension. After extracting everything into 2 folders, “before delete” and “after delete”, the folders were added to AccessData FTK as live evidence.

4 Analysis & Findings

After extraction, both files had the same structure. The backed up files were divided into 3 main tar files: ***App.tar***, ***Setting.tar*** and ***Media.tar***. Along with these tarballs was an xml file describing the content of the files called ***Manifest.xml***. It showed the device PIN and OS version as well as file size for the above mentioned tarballs as shown in Figure 1.



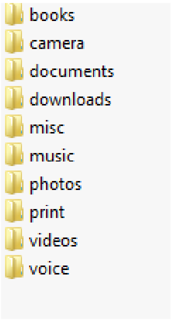
```
  <?xml version="1.0" encoding="UTF-8" ?>
- <BlackBerry_Backup>
    <Client platform="windows" osversion="Microsoft Windows NT 6.1.7601 Service Pack 1" dtmversion="6.1.0.35" />
    <Version>2.0</Version>
    <Encryption type="RIM_AES_CBC" version="1.0" Salt="" />
  - <SourceDevice pin="50109FDE" hwid="6001A06">
      <Platform type="QNX" version="2.0.0.7971" />
    </SourceDevice>
  - <QnxOSDevice>
    - <Archives>
        <Archive id="app" name="Application Data" count="71" bytesize="21430784" />
        <Archive id="media" name="Media" count="20" bytesize="8634368" />
        <Archive id="settings" name="Settings" count="820" bytesize="1114624" />
      </Archives>
    </QnxOSDevice>
  </BlackBerry_Backup>
```

Fig. 1. Content of Manifest.xml

4.1 Media.tar

Examining tar files using WinRAR, we first started out with the ***Media.tar*** file which had two folders in it, ***Media*** and ***dtm***. The "Media" folder has the same structure of folders when you connect your device to the PC as shown in Figure 2.

We found all the images as well as the video taken by the camera in the folder ***Camera*** in the first .bbb file. Aditionally, all the uploaded images were saved in the***\photos\Pictures\BlackBerry*** folder. There were no traces of the deleted image taken by the camera, but moving into the ***dtm*** folder of the "after delete" .bbb file we found the file ***c2f39ce100000004.bbms*** which listed the file name of all images uploaded into the device including the one we deleted.



books
camera
documents
downloads
misc
music
photos
print
videos
voice

Fig. 2. Media folder content

4.2 Settings.tar

Settings.tar is an archived folder containing several files. Notably, ***Settings.tar*** contained another file called ***dynamic.lm*** in the directory ***\accounts\1000\sys\input\fluency\user***. This file contained the emails sent from the device.

The directory ***\pps\services*** listed all the services in the device such as: accelerometer, audio, clock, geolocation, input, light_sensor and more. Table 1 summarises the files found with evidence in them:

Table 1. Evidence Files in services folder

File name	Path	Description
Status	\pps\services\accelerometer	The file shows the orientation of the device at the time of backup, and whether it was facing up or down.

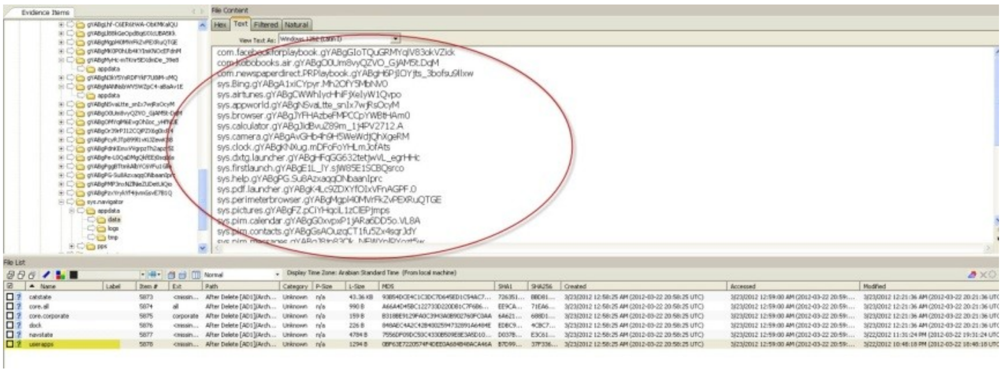
File name	Path	Description
Status	\pps\services\audio	The file shows the audio status and whether a2dp bluetooth audio is enabled or not.
Status	\pps\services\clock	The file showed which time zone the device was using.
Status	\pps\services\geolocation \country	The file showed the country code for the country the device was in.
Status	\pps\services\network-time	Showed the time stamp of the clock update and the ntp server used.
orientation	\pps\services\sensor	Same information provided by the accelerometer status file
Settings	\pps\system	The file contained: • Time format • Langauge used • Time Zone

Notably we found two sub-folders in the folder ***\settings\var*** which appear worthy of further investigation; ***certmgr*** and ***keymgr***. The first one seemed to contain all the certificates the device uses for communication and the other one contained a set of private keys.

Digging further in the folder we found the file ***wpa_pps.conf*** in the directory ***\var\etc\netsecure*** which stored all the info related to the wireless networks to which the device had been connected. Another interesting finding was that the device also copied all the networks to which the BlackBerry smart phone had ever connected, including all the SSIDs and passwords, in clear text. This included wireless networks to which the BlackBerry smart phone had connected before it had connected to the PlayBook device using BlackBerry Bridge.

4.3 Apps.tar

This file contained obscured folder names, similar to what Apple does with iOS application folders with obfuscated names. We speculate that the names may be generated through a hash function of some description. Within Apps.tar, we found a file named ***core.all*** in the directory ***sys.navigator\appdata\data***. This file can be thought of as a map for the obfuscated folders within the tarball. Furthermore, in the same folder were other files that were a subset of ***core.all***, like ***userapps*** (shown in Figure 3) which lists only third party apps installed, and ***core.corporate*** that lists all the OS built-in apps, while the file dock showed the apps "pinned" to the dock in the PlayBook's GUI.



View Text As: Windows 1252 (Latin 1)

```
com.facebookforplaybook.gYABgGIoTQuGRMYqlV83okVZick
com.kobobooks.air.gYABgO0Um8vyQZVO_GjAM5t.DqM
com.newspaperdirect.PRPlaybook.gYABgH6PjIOYjts_3bofsu9llxw
sys.Bing.gYABgA1xiCYpyr.Mh2OfY5MbNV0
sys.airtunes.gYABgCWWhlydHhiFjXelyW1Qvpo
sys.appworld.gYABgNSvaLtte_snIx7wjRsOcyM
sys.browser.gYABgJYFHAzbeFMPCCpYWBtHAm0
sys.calculator.gYABgJidBvuZ89m_1j4PV2712.A
sys.camera.gYABgAvGHb4h9H5WeWdjQhXgeRM
sys.clock.gYABgKNXug.mDFoFoYHLmJofAts
sys.dxtg.launcher.gYABgHFqGG632tetjwVL_egrHHc
sys.firstlaunch.gYABgE1L_lY.sjW85E1SCBQsrco
sys.help.gYABgPG.SuBAzxaqqONbaanIprc
sys.pdf.launcher.gYABgK4Lc9ZDXYfOIxVFnAGPF.0
sys.perimeterbrowser.gYABgMgpI40MVrFkZvPEXRuQTGE
sys.pictures.gYABgFZ.pCIYHqcIL1zCIEPjmps
sys.pim.calendar.gYABgG0xvpxP1jARa6DD5o.VL8A
sys.pim.contacts.gYABgGsAOuzqCT1fu5Zx4sqrJdY
```

Normal
Display Time Zone: Arabian Standard Time (From local machine)

Name	Label	Item #	Ext	Path	Category	P-Size	L-Size	MD5	SHA1	SHA256	Created	Accessed	Modified
cabstate		5873	<missin...	After Delete [AD1]\Arch...	Unknown	n/a	43.36 KB	93B54DCE4C1C3DC7D645ED1C54AC7...	726351...	8BD81...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/23/2012 12:21:36 AM (2012-03-22 20:21:36 UTC
core.all		5874	all	After Delete [AD1]\Arch...	Unknown	n/a	990 B	A66A4D45BC122733D220D81C7F686...	EE9CA...	71EA6...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/23/2012 12:21:36 AM (2012-03-22 20:21:36 UTC
core.corporate		5875	corporate	After Delete [AD1]\Arch...	Unknown	n/a	159 B	B318BE9129FA0C3943A0B902760FC0AA	6A621...	688D1...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/23/2012 12:21:36 AM (2012-03-22 20:21:36 UTC
dock		5876	<missin...	After Delete [AD1]\Arch...	Unknown	n/a	226 B	848AEC4A2C4284002594732891A6484E	EDBC9...	4CBC7...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/23/2012 12:21:36 AM (2012-03-22 20:21:36 UTC
navstate		5877	<missin...	After Delete [AD1]\Arch...	Unknown	n/a	4784 B	7556DF09DC50C4330B509E8E3A6D10...	D037B...	E3C61...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/22/2012 11:31:24 PM (2012-03-22 19:31:24 UTC
userapps		5878	<missin...	After Delete [AD1]\Arch...	Unknown	n/a	1294 B	0BF63E7220574F4DEE0A684B46ACA46A	B7D99...	37F336...	3/23/2012 12:58:25 AM (2012-03-22 20:58:25 UTC)	3/23/2012 12:59:00 AM (2012-03-22 20:59:...	3/22/2012 10:48:18 PM (2012-03-22 18:48:18 UTC

Fig. 3. userapps lists all third party apps installed on the BlackBerry PlayBook device

Using the abovementioned files we focused on the folders of apps that may have potential evidence in them. We started by examining the browser's folder as we expected it to be the richest in terms of recoverable data. The browser's folder was named ***gYABgJYFHAzbeFMPCCpYWBtHAm0*** and it contained the files shown in Table 2.

Table 2. Evidence from browser app

File name	Path	Description
settings.sol	\#SharedObjects \browser.swf	This file showed the settings used by the browser: history expiry, homepage, default search engine, encoding used, font size and user agent string can be found.
Cache(folder)	\appdata\data\cache	This folder contained cached web files which can be used to reconstruct browsing history and browsed pages.

File name	Path	Description
WebpageIcons.db	\appdata\data\database	An SQLite database that contains information about visted sites' fav icon and where to get them. Can be used to track browsing histroy.
Favicon(folder)	\appdata\data\favicon	Fav icon are cashed in this folder in png formats.
Snapshot(folder)	\appdata\data\snapshot	This folder holds the snapshots of the visited websites as seen on the browser history section. It stores them in 2 sizes in landscape and portrait.
browser-v1.0.db	\appdata\data	A SQLite database that has 2 tables, Bookmark and history, which are self explanatory.
cookieCollection.db	\appdata\data	Another SQLite database that stores all the cookies that are stored on the device with information like: host, expiry and last accessed

Further key term searching led us to the YouTube folder ***gYABgPcyRJTp899l1vKiJZewK88***, to the file ***qnx.youtube.sol*** located in ***\appdata\data\#SharedObjects\Youtube.swf***. Here we also found additional information about the clip we watched, including the URL, the URL of the comments and some related videos. The folder ***appdata\data\appdata*** had a SQLite database, ***cookies.sqlite***, that held the cookies used for the YouTube application and others.

4.7 Limitations

The analysis was conducted on only one PlayBook device, so we couldn't record how hardware changes might affect the results, if indeed they would affect them. The experiment was performed on the original PlayBook device, not the new 4G LTE PlayBook device, which is capable of connecting to the mobile network independent of the BlackBerry Bridge tethering feature. Likewise; our examination of artifacts left as a result of the BlackBerry Bridge tethering feature only involved one additional device, a BlackBerry Bold 9900. Other BlackBerry smart phone devices were not used in this experiment, although BlackBerry Bridge is supported on a wide range of

BlackBerry smart phone models. Most significantly, the technique described in this paper depends on logical acquisition through the BlackBerry backup procedure, and therefore shares the limitations common to logical acquisitions of all digital devices. Our plans to address these limitations are discussed in section 5, below.

5 Conclusions and Future Work

The original (non-4G LTE) BlackBerry PlayBook device is a low-priced tablet which integrates with the BlackBerry smart phone device. Despite an overall smaller share of the tablet and smart phone markets than iOS and Android-based competitors, the BlackBerry devices (PlayBook and smart phones alike) remain popular and widely deployed in the corporate and government markets, and in the mainstream consumer market in many countries. This paper described a logical acquisition-based approach to investigating the BlackBerry PlayBook device. Our approach is based on the use of the BlackBerry PlayBook backup file created by the normal backup procedure through the BlackBerry Desktop Management software. We examined the backup data structure and identified files stored within which appeared to be of forensic interest. Table 3 lists the files within this backup structure which we identified as likely containing information of interest to a digital investigation.

Table 3. Summary of results

Tarball	File Path within Tarball	Description
Settings.tar	\accounts\1000\sys\input\fluency\user \dynamic.lm	Emails that are sent from the device.
Settings.tar	\pps\services\accelerometerb\Status	Orientation of the device at the time of the backup
Settings.tar	\pps\services\audio\Status	Bluetooth and A2DP usage
Settings.tar	\pps\services\clock\Status	Current Time Zone
Settings.tar	\pps\services\geolocation\country \Status	Country code for geo location at the time of the backup
Settings.tar	\pps\services\network-time\Status	NTP server used and time stamp for last update (Unix EPOCH time)
Settings.tar	\pps\services\sensor\orientation	Orientation of the device at the time of the backup
Settings.tar	\pps\system\Settings	Language, time format and time zone
Settings.tar	\settings\var 2\certmgr	x.509 certificates

Tarball	File Path within Tarball	Description
Settings.tar	\settings\var 2\keymgr	Private keys
Settings.tar	\var\etc\netsecure\wpa_pps.conf	Information about wireless networks, including passwords
Media.tar	\media\camera	All the images and videos taken by the camera, none of the deleted
Media.tar	\dtm\MediaSync\c2f39ce100000004.bbms	List of all images synced to the device, even the deleted ones.
Apps.tar	\sys.navigator\appdata\data\core.all	Map to all application folders
Apps.tar	\sys.navigator\appdata\data\core.corporate	Subset that shows only OS built in applications
Apps.tar	\sys.navigator\appdata\data\userapps	Subset that shows user installed applications
Apps.tar	\gYABgJYFHAzbeFMPCCpYWBtHAm0\	Browser application folder, can be different
Apps.tar	\SharedObjects\browser.swf \settings.sol	Browser settings
Apps.tar	\appdata\data\cache\	Browser Cache
Apps.tar	\appdata\data\database \WebpageIcons.db	Fav icon information
Apps.tar	\appdata\data\favicon\	Fav icon image files
Apps.tar	\appdata\data\snapshot	Websites snapshots
Apps.tar	\appdata\data\ browser-v1.0.db	Bookmarks and history tables
Apps.tar	\appdata\data\cookieCollection.db	Browser cookies
Apps.tar	\gYABgE1L_lY-sjW85E1SCBQsrco \firstlaunch.sol	Device name and BlackBerry ID used to initiate the device
Apps.tar	\gYABgPcyRJTp899l1vKiJZewK88	YouTube Application folder
Apps.tar	\appdata\data\#SharedObjects \Youtube.swf	YouTube searches and videos watched.
Apps.tar	\appdata\data\appdata\cookies.sqlite	Cookies stored by YouTube Application.

In the future, we plan to run more extensive tests on a broader range of BlackBerry hardware and software. We also plan on creating stronger usage scenarios to create more complete user profiles. As we continue to investigate these devices, it will be possible to develop a software parser for the PlayBook backup structure which can be used to automate the discovery of the different items of interest we discovered in the investigation described in this paper. This parser could be combined with a convenient user interface to display or export this information, to assist forensic investigators.

The new 4G LTE BlackBerry PlayBook is substantially similar to the original BlackBerry PlayBook except for the addition of a 4G LTE modem. Although this new model still supports BlackBerry Bridge, the 4G LTE modem allows connection to the mobile network directly through a micro-SIM. We anticipate that the back-up structure would be extremely similar to the structure described in this paper, but have yet to confirm this through our own testing. One point of interest is that the new 4G LTE modem may lead to a significant reduction in the use of the BlackBerry Bridge feature which, as we have shown, leaves interesting evidence about paired BlackBerry smart phones used with the subject PlayBook.

Another limitation of the work described in this paper, as noted above, is our dependence on the BlackBerry PlayBook backup file. The backup file may be thought of as a logically acquired image of the PlayBook device, and as with all logical acquisitions, there may be some additional evidence stored on the device itself which cannot be retrieved. For example, deleted files or data stored in primary memory only as opposed to secondary storage on the device (e.g. cryptographic keys, passphrases) may be of forensic interest, but will not be retrieved through a logical acquisition of the PlayBook device's secondary storage. We plan to address this deficiency by investigating techniques for physical acquisition of the PlayBook device. DingleBerry [17] is a PlayBook hacking tool which permits root access to the BlackBerry PlayBook device. DingleBerry may provide a mechanism for our future work in the physical acquisition of the BlackBerry PlayBook device.

References

1. Gartner Research. (2012). *Gartner Says Worldwide Media Tablets Sales to Reach 119 Million Units in 2012*. Retrieved from http://www.gartner.com/it/page.jsp?id=1980115
2. *BlackBerry PlayBook cleared for government use.* Retrieved from http://www.cbc.ca/news/technology/story/2011/07/22/technology-BlackBerry-PlayBook-rim.html
3. Ali, S., AlHosani, S., AlZarooni, F. and Baggili, I. (2012). iPad2 logical acquisition: Automated or manual examination? *Proceedings of the 2012 ADFSL Conference on Digital Forensics, Security and Law*. Richmond, VA.
4. Garfinkel, S. L. (2010). Digital forensics research: The next 10 years. *Proceedings of the 2010 Digital Forensics Workshop* published in *Digital Investigation*, Vol. 7, S64-S73. doi:10.1016/j.diin.2010.05.009
5. Bader, M. and Baggili, I. (2010). iPhone 3GS Forensics: Logical analysis using Apple iTunes Backup Utility. *Small Scale Digital Device Forensics Journal*, vol. 4(1).
6. Gómez-Miralles, L., and Arnedo-Moreno, J. (2011), Universal, Fast Method for iPad Forensics Imaging via USB Adapter., *Fifth International Conference on Innovative*

Mobile and Internet Services in Ubiquitous Computing (IMIS). (pp.200-207). Valencia.

7. Iqbal, B., Iqbal, A., and Al Obaidli, H. (2012), A Novel Method of iDevice (iPhone, iPad, iPod) Forensics without Jailbreaking., *2012 International Conference on Innovations in Information Technology (IIT).* Al Ain.
8. Lessard, J., & Kessler, G. C. (2010). Android Forensics: Simplifying Cell Phone Examinations. *Small Scale Digital Device Forensics Journal*, vol. 4(1).
9. Vidas, T., Zhang, C., & Christin, N. (2011). Toward a general collection methodology for Android devices. *Proceedings of the 2011 Digital Forensics Workshop* published in *Digital Investigation*, Vol. 8, S14-S24.
10. Valli, C. and Jones, A. (2008). A Study into the Forensic Recoverability of Data from 2nd Hand BlackBerry Devices: World-Class Security, Foiled by Humans. *Proceedings of World Congress in Computer Science, Computer Engineering and Applied Computing.*(pp.604-607). Las Vegas.
11. Hoog A., Strzempka K. (2010). *Independent Research and Reviews of iPhone Forensic Tools*. Retrieved from https://viaforensics.com/resources/white-papers/iphone-forensics/
12. National Institute of Standards and Technology (2010), *Test Results for Mobile Device Acquisition Tool: Zdziarski's Method*, http://www.nij.gov/pubs-sum/232383.htm
13. Gómez-Miralles, L., & Arnedo-Moreno, J. (2012), Versatile iPad forensic acquisition using the Apple Connection Kit. *Computers & Mathematics with Applications,* vol. 63, no.2, pp544-553.
14. Golubev, N. (October 28, 2011), *Android Forensics Study of Password and Pattern Lock Protection* retrieved from http://android-forensics.com/android-forensics-study-of-password-and-pattern-lock-protection/143
15. Thing, V. L. L., Kian-Yong Ng, Ee-Chien Chang. (2010), Live memory forensics of mobile phones, *2010 Digital Forensics Research Workshop* published in *Digital Investigation*, vol. 7, S74-S82.
16. Sylve J., Case A., Marziale L., Richard G.G. (2012), Acquisition and analysis of volatile memory from Android devices, *Digital Investigation,* vol. 8, no.3-4, pp175-184.
17. Wade, C. (2012), http://www.dingleberry.it/, visited 5 July 2012.